

9.0

Managed File Transfer

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 971.

Cette édition s'applique à la version 9 édition 0 d' IBM® MQ et à toutes les éditions et modifications ultérieures, sauf indication contraire dans les nouvelles éditions.

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

© **Copyright International Business Machines Corporation 2007, 2023.**

Table des matières

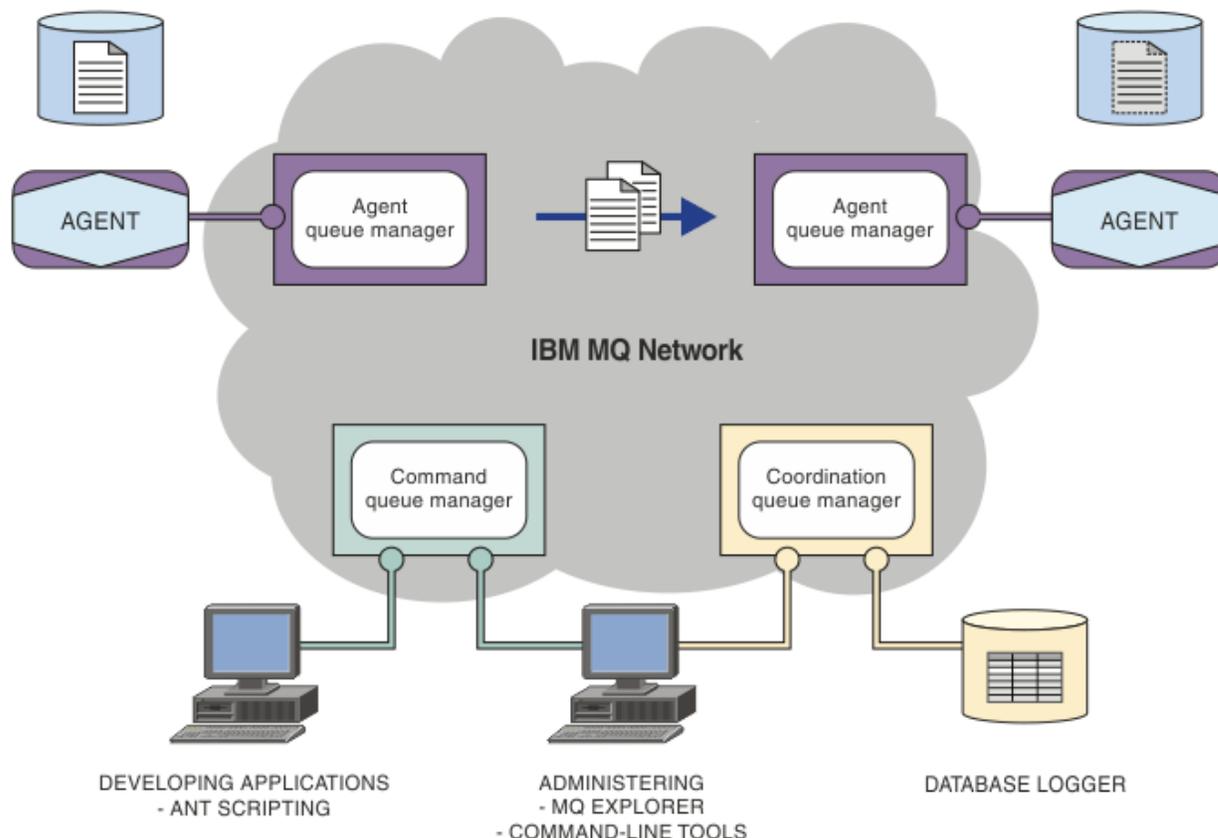
Managed File Transfer.....	5
Présentation de MFT.....	7
Introduction MFT.....	7
Fonctionnement de MFT avec IBM MQ.....	10
Présentation de la topologie MFT.....	10
Présentation de MFT REST API.....	11
Nouveautés et modifications apportées à MFT dans la IBM MQ 9.0.....	12
Remarques sur la migration pour MFT.....	13
MFT Présentation du scénario.....	29
Topologies courantes de MFT.....	29
Configuration du serveur de base.....	33
Sécurisation de MFT.....	41
Authentification de connexion MFT et IBM MQ.....	42
MFT bacs à sable.....	46
Configuration du chiffrement SSL ou TLS pour MFT.....	52
Connexion à un gestionnaire de files d'attente en mode client avec authentification de canal.....	54
Configuration de SSL ou TLS entre l'agent de pont Connect:Direct et le noeud Connect:Direct.....	55
configurationMFT.....	57
Options de configuration de MFT sur Multiplatforms.....	58
Options de configuration MFT sous z/OS.....	60
Configuration de l'Redistributable Managed File Transfer Agent.....	60
Création d'un fichier de commandes d'agent ou de consignateur MFT.....	64
Configuration de Managed File Transfer for z/OS.....	65
Configuration de MFT sur IBM i.....	95
Configuration de MFT pour la première utilisation.....	97
Configuration d'un consignateur MFT.....	107
Configuration du pont Connect:Direct.....	146
Configuration des agents MFT avec MSCS.....	156
Administration d'MFT.....	157
Démarrage d'un agent MFT.....	158
Démarrage d'un nouveau transfert de fichiers.....	163
Création d'un transfert de fichiers planifié.....	166
Utilisation des transferts de fichiers en attente.....	168
Déclenchement d'un transfert de fichiers.....	168
Surveillance des transferts de fichiers en cours.....	170
Affichage du statut des transferts de fichiers dans le journal de transfert.....	172
Surveillance des ressources MFT.....	174
Utilisation des modèles de transfert de fichiers.....	200
Transfert de données de fichiers vers des messages.....	203
Transfert de données de messages vers des fichiers.....	211
Liste des agents MFT.....	217
Arrêt d'un agent MFT.....	218
Le pont de protocole.....	219
Le pont Connect:Direct.....	236
Utilisation de MFT à partir de IBM Integration Bus.....	252
MFT Reprise et redémarrage.....	252
Option de délai d'attente pour les transferts de fichiers en cours de récupération.....	253
Développement d'applications pour MFT.....	256
Spécification des programmes à exécuter avec MFT.....	256
Utilisation d' Apache Ant avec MFT.....	258
Personnalisation de MFT avec des exits utilisateur.....	263
Contrôle de MFT via le placement de messages dans la file d'attente de commandes d'agent.....	276

traitement des incidentsMFT.....	277
MFT identification des problèmes généraux.....	277
Traitement des incidents liés au pont Connect:Direct.....	353
Référence MFT.....	359
Référence de sécurité MFT.....	359
Les commandes et processus MFT qui se connectent à quel gestionnaire de files d'attente.....	376
Commandes MFT.....	383
Référence de configuration MFT.....	547
Référence d'administration MFT.....	642
Référence des applications de développement MFT.....	869
Messages de diagnostic MFT.....	970
Remarques.....	971
Documentation sur l'interface de programmation.....	972
Marques.....	972

Managed File Transfer

Managed File Transfer transfère des fichiers entre des systèmes d'une manière gérée et auditable, quelle que soit la taille des fichiers ou des systèmes d'exploitation utilisés.

Vous pouvez utiliser Managed File Transfer pour développer une solution automatisée, évolutive et personnalisée qui vous permette de gérer, d'approuver et de sécuriser les transferts de fichiers. Managed File Transfer élimine les redondances coûteuses, diminue les coûts de maintenance et maximise vos investissements informatiques existants.



Le diagramme illustre une topologie Managed File Transfer simple. Il existe deux agents, chacun se connectant à son propre gestionnaire de file d'attente d'agent sur un réseau IBM MQ. Un fichier est transféré de l'agent d'un côté du diagramme, par l'intermédiaire du réseau IBM MQ, vers l'agent de l'autre côté du diagramme. En outre, le réseau IBM MQ comporte le gestionnaire de file d'attente de coordination et un gestionnaire de file d'attente de commandes. Les applications et les outils se connectent à ces gestionnaires de files d'attente pour configurer, administrer, exploiter et consigner l'activité Managed File Transfer dans le réseau IBM MQ.

Managed File Transfer peut être installé avec quatre options différentes, en fonction de votre système d'exploitation et de la configuration générale. Ces options sont Managed File Transfer Agent, Managed File Transfer Logger, Managed File Transfer Service et Managed File Transfer Tools. Pour plus d'informations, voir [Options du produit Managed File Transfer](#).

Vous pouvez utiliser Managed File Transfer pour effectuer les tâches suivantes :

- Créer des transferts de fichiers gérés
 - Créer des transferts de fichiers à partir d'IBM MQ Explorer sur des plateformes Linux® ou Windows.

- Créer des transferts de fichiers à partir de la ligne de commande sur toutes les plateformes prises en charge.
- Intégrer la fonction de transfert de fichiers dans l'outil Apache Ant.
- Ecrire des applications qui contrôlent Managed File Transfer en plaçant les messages sur les files d'attente des commandes d'agent.
- Planifier les transferts de fichiers à un moment ultérieur. Vous pouvez également déclencher des transferts de fichiers planifiés en fonction d'une plage d'événements du système de fichiers (par exemple, création d'un fichier).
- Surveiller en permanence une ressource, telle qu'un répertoire, et, lorsque le contenu de cette ressource satisfait une condition prédéfinie, lancer une tâche. Il peut s'agir d'un transfert de fichiers, d'un script Ant ou d'un travail JCL.
- Transférer des fichiers vers et depuis des files d'attente IBM MQ.
- Transférer des fichiers vers et depuis des serveurs FTP, FTPS ou SFTP.
- Transfert de fichiers vers et depuis des noeuds Connect:Direct.
- Transférer des fichiers texte et binaires. Les fichiers texte sont convertis automatiquement entre les pages de code et les conventions de fin de ligne des systèmes source et cible.
- Les transferts peuvent être sécurisés, à l'aide des normes de l'industrie pour les connexions SSL (Secure Socket Layer).
- Afficher les transferts en cours et les informations de journal sur tous les transferts sur votre réseau.
 - Afficher le statut des transferts en cours d'IBM MQ Explorer sur des plateformes Linux ou Windows.
 - Vérifier le statut des transferts terminés à l'aide d'IBM MQ Explorer sur des plateformes Linux ou Windows.
 - Utiliser la fonction du consignateur de base de données Managed File Transfer pour sauvegarder les messages de journal dans une base de données Db2 ou Oracle.

Managed File Transfer est développé sur IBM MQ, qui permet une distribution ponctuelle et garantie des messages entre les applications. Vous pouvez profiter des diverses fonctions d'IBM MQ. Par exemple, vous pouvez utiliser la compression des canaux pour compresser les données que vous envoyez entre les agents via les canaux IBM MQ et utiliser les canaux SSL pour sécuriser les données que vous envoyez entre les agents. Les fichiers sont transférés de manière fiable et peuvent tolérer un incident de l'infrastructure par l'intermédiaire de laquelle le transfert de fichiers est effectué. En cas de panne réseau, le transfert de fichiers redémarre à partir de là où il s'était arrêté lorsque la connectivité est restaurée.

En consolidant le transfert de fichiers avec votre réseau IBM MQ existant, vous pouvez éviter d'utiliser les ressources requises pour gérer deux infrastructures distinctes. Si vous n'êtes pas déjà un client d'IBM MQ, en créant un réseau IBM MQ pour prendre en charge Managed File Transfer, vous développez le réseau principal d'une future implémentation d'architecture SOA. Si vous êtes déjà un client d'IBM MQ, Managed File Transfer peut profiter de votre infrastructure IBM MQ et notamment d'IBM MQ Internet Pass-Thru et d'IBM Integration Bus.

Managed File Transfer s'intègre à un certain nombre d'autres produits IBM :

IBM Integration Bus

Fichiers de processus transférés par Managed File Transfer dans le cadre d'un flux IBM Integration Bus. Pour plus d'informations, voir la section [«Utilisation de MFT à partir de IBM Integration Bus»](#), à la page 252.

IBM Sterling Connect:Direct

Transférez des fichiers vers et depuis un réseau Connect:Direct existant à l'aide d'un pont Managed File Transfer Connect:Direct. Pour plus d'informations, voir la section [«Le pont Connect:Direct»](#), à la page 236.

IBM Tivoli Composite Application Manager

IBM Tivoli Composite Application Manager fournit un agent que vous pouvez utiliser pour surveiller les informations publiées sur le gestionnaire de file d'attente de coordination.

Concepts associés

[«Présentation de la topologie MFT», à la page 10](#)

Présentation de la manière dont les agents Managed File Transfer sont connectés au gestionnaire de file d'attente de coordination dans un réseau IBM MQ.

[«Fonctionnement de MFT avec IBM MQ», à la page 10](#)

Managed File Transfer interagit de diverses façons avec IBM MQ.

Information associée

[Options du produit Managed File Transfer](#)

Présentation de MFT

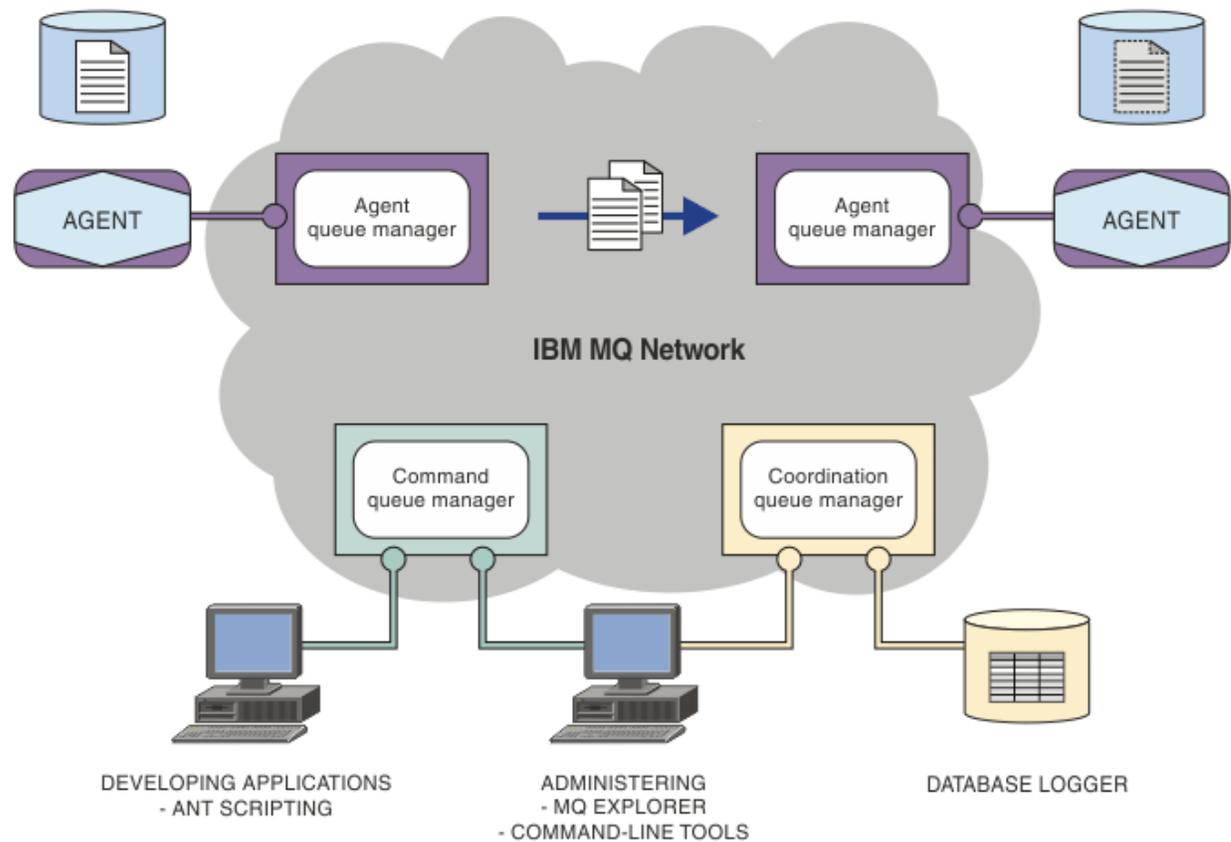
Cette section contient des informations de présentation dont vous pouvez vous servir pour vous initier à Managed File Transfer.

- [«Managed File Transfer», à la page 5](#)
- [Options du produit Managed File Transfer](#)
- [«Présentation de la topologie MFT», à la page 10](#)
- [«Nouveautés et modifications apportées à MFT dans la IBM MQ 9.0», à la page 12](#)

Managed File Transfer

Managed File Transfer transfère des fichiers entre des systèmes d'une manière gérée et auditable, quelle que soit la taille des fichiers ou des systèmes d'exploitation utilisés.

Vous pouvez utiliser Managed File Transfer pour développer une solution automatisée, évolutive et personnalisée qui vous permette de gérer, d'approuver et de sécuriser les transferts de fichiers. Managed File Transfer élimine les redondances coûteuses, diminue les coûts de maintenance et maximise vos investissements informatiques existants.



Le diagramme illustre une topologie Managed File Transfer simple. Il existe deux agents, chacun se connectant à son propre gestionnaire de file d'attente d'agent sur un réseau IBM MQ. Un fichier est transféré de l'agent d'un côté du diagramme, par l'intermédiaire du réseau IBM MQ, vers l'agent de l'autre côté du diagramme. En outre, le réseau IBM MQ comporte le gestionnaire de file d'attente de coordination et un gestionnaire de file d'attente de commandes. Les applications et les outils se connectent à ces gestionnaires de files d'attente pour configurer, administrer, exploiter et consigner l'activité Managed File Transfer dans le réseau IBM MQ.

Managed File Transfer peut être installé avec quatre options différentes, en fonction de votre système d'exploitation et de la configuration générale. Ces options sont Managed File Transfer Agent, Managed File Transfer Logger, Managed File Transfer Service et Managed File Transfer Tools. Pour plus d'informations, voir [Options du produit Managed File Transfer](#).

Vous pouvez utiliser Managed File Transfer pour effectuer les tâches suivantes :

- Créer des transferts de fichiers gérés
 - Créer des transferts de fichiers à partir d'IBM MQ Explorer sur des plateformes Linux ou Windows.
 - Créer des transferts de fichiers à partir de la ligne de commande sur toutes les plateformes prises en charge.
 - Intégrer la fonction de transfert de fichiers dans l'outil Apache Ant.
 - Ecrire des applications qui contrôlent Managed File Transfer en plaçant les messages sur les files d'attente des commandes d'agent.
 - Planifier les transferts de fichiers à un moment ultérieur. Vous pouvez également déclencher des transferts de fichiers planifiés en fonction d'une plage d'événements du système de fichiers (par exemple, création d'un fichier).
 - Surveiller en permanence une ressource, telle qu'un répertoire, et, lorsque le contenu de cette ressource satisfait une condition prédéfinie, lancer une tâche. Il peut s'agir d'un transfert de fichiers, d'un script Ant ou d'un travail JCL.

- Transférer des fichiers vers et depuis des files d'attente IBM MQ.
- Transférer des fichiers vers et depuis des serveurs FTP, FTPS ou SFTP.
- Transfert de fichiers vers et depuis des noeuds Connect:Direct.
- Transférer des fichiers texte et binaires. Les fichiers texte sont convertis automatiquement entre les pages de code et les conventions de fin de ligne des systèmes source et cible.
- Les transferts peuvent être sécurisés, à l'aide des normes de l'industrie pour les connexions SSL (Secure Socket Layer).
- Afficher les transferts en cours et les informations de journal sur tous les transferts sur votre réseau.
 - Afficher le statut des transferts en cours d'IBM MQ Explorer sur des plateformes Linux ou Windows.
 - Vérifier le statut des transferts terminés à l'aide d'IBM MQ Explorer sur des plateformes Linux ou Windows.
 - Utiliser la fonction du consignateur de base de données Managed File Transfer pour sauvegarder les messages de journal dans une base de données Db2 ou Oracle.

Managed File Transfer est développé sur IBM MQ, qui permet une distribution ponctuelle et garantie des messages entre les applications. Vous pouvez profiter des diverses fonctions d'IBM MQ. Par exemple, vous pouvez utiliser la compression des canaux pour compresser les données que vous envoyez entre les agents via les canaux IBM MQ et utiliser les canaux SSL pour sécuriser les données que vous envoyez entre les agents. Les fichiers sont transférés de manière fiable et peuvent tolérer un incident de l'infrastructure par l'intermédiaire de laquelle le transfert de fichiers est effectué. En cas de panne réseau, le transfert de fichiers redémarre à partir de là où il s'était arrêté lorsque la connectivité est restaurée.

En consolidant le transfert de fichiers avec votre réseau IBM MQ existant, vous pouvez éviter d'utiliser les ressources requises pour gérer deux infrastructures distinctes. Si vous n'êtes pas déjà un client d'IBM MQ, en créant un réseau IBM MQ pour prendre en charge Managed File Transfer, vous développez le réseau principal d'une future implémentation d'architecture SOA. Si vous êtes déjà un client d'IBM MQ, Managed File Transfer peut profiter de votre infrastructure IBM MQ et notamment d'IBM MQ Internet Pass-Thru et d'IBM Integration Bus.

Managed File Transfer s'intègre à un certain nombre d'autres produits IBM :

IBM Integration Bus

Fichiers de processus transférés par Managed File Transfer dans le cadre d'un flux IBM Integration Bus. Pour plus d'informations, voir la section [«Utilisation de MFT à partir de IBM Integration Bus»](#), à la page 252.

IBM Sterling Connect:Direct

Transférez des fichiers vers et depuis un réseau Connect:Direct existant à l'aide d'un pont Managed File Transfer Connect:Direct. Pour plus d'informations, voir la section [«Le pont Connect:Direct»](#), à la page 236.

IBM Tivoli Composite Application Manager

IBM Tivoli Composite Application Manager fournit un agent que vous pouvez utiliser pour surveiller les informations publiées sur le gestionnaire de file d'attente de coordination.

Concepts associés

[«Présentation de la topologie MFT»](#), à la page 10

Présentation de la manière dont les agents Managed File Transfer sont connectés au gestionnaire de file d'attente de coordination dans un réseau IBM MQ.

[«Fonctionnement de MFT avec IBM MQ»](#), à la page 10

Managed File Transfer interagit de diverses façons avec IBM MQ.

Information associée

[Options du produit Managed File Transfer](#)

Fonctionnement de MFT avec IBM MQ

Managed File Transfer interagit de diverses façons avec IBM MQ.

- Managed File Transfer transfère des fichiers entre des processus d'agent en divisant chaque fichier en un ou plusieurs messages et en transmettant les messages sur votre réseau IBM MQ.
- Les processus d'agent déplacent les données de fichier à l'aide de messages non persistants afin de limiter l'impact sur vos journaux IBM MQ. En communiquant les uns avec les autres, les processus d'agent régulent le flux de messages contenant des données de fichier. Ainsi, les messages contenant des données de fichier ne s'accumulent pas dans les files d'attente de transmission IBM MQ et au cas où un message non persistant ne serait pas distribué, les données de fichier sont envoyées à nouveau.
- Les agents Managed File Transfer utilisent plusieurs files d'attente IBM MQ. Pour plus d'informations, voir la section «[Les files d'attente système MFT et la rubrique système](#)», à la page 639.
- Bien que certaines de ces files d'attente soient strictement réservées à un usage interne, un agent peut accepter des demandes sous forme de messages de commande formatés spécialement, qui sont envoyés dans une file d'attente spécifique lue par l'agent. Les commandes de ligne de commande et le plug-in IBM MQ Explorer envoient des messages IBM MQ à l'agent pour l'inviter à effectuer l'action voulue. Vous pouvez écrire des applications IBM MQ qui interagissent avec l'agent de cette façon. Pour plus d'informations, voir la section «[Contrôle de MFT via le placement de messages dans la file d'attente de commandes d'agent](#)», à la page 276.
- Les agents Managed File Transfer envoient des informations sur leur état et la progression et le résultat des transferts vers un gestionnaire de files d'attente MQ qui a été désigné comme gestionnaire de file d'attente de coordination. Ces informations sont publiées par le gestionnaire de file d'attente de coordination et les applications souhaitant surveiller la progression des transferts ou conserver des enregistrements des transferts effectués peuvent s'y abonner. Les commandes de ligne de commande et le plug-in IBM MQ Explorer peuvent utiliser les informations qui sont publiées. Vous pouvez écrire des applications IBM MQ qui utilisent ces informations. Pour plus d'informations sur la rubrique dans laquelle les informations sont publiées, voir «[Rubrique SYSTEM.FTE](#)», à la page 636.
- Les composants principaux de Managed File Transfer bénéficient de la capacité des gestionnaires de files d'attente IBM MQ à stocker et retransmettre des messages. En d'autres termes, en cas d'indisponibilité, les parties non affectées de votre infrastructure peuvent continuer de transférer des fichiers. Cette possibilité s'étend au gestionnaire de file d'attente de coordination, où une combinaison de stockage et de retransmission et d'abonnements durables permet au gestionnaire de file d'attente de coordination d'être indisponible sans que des informations essentielles sur les transferts de fichiers qui ont eu lieu ne soient perdues.

Présentation de la topologie MFT

Présentation de la manière dont les agents Managed File Transfer sont connectés au gestionnaire de file d'attente de coordination dans un réseau IBM MQ.

Les agents Managed File Transfer envoient et reçoivent les fichiers transférés. Chaque agent possède son propre ensemble de files d'attente sur son gestionnaire de files d'attente associé et l'agent est connecté à son gestionnaire de files d'attente en mode liaisons ou client. Il peut également utiliser le gestionnaire de file d'attente de coordination comme gestionnaire de files d'attente.

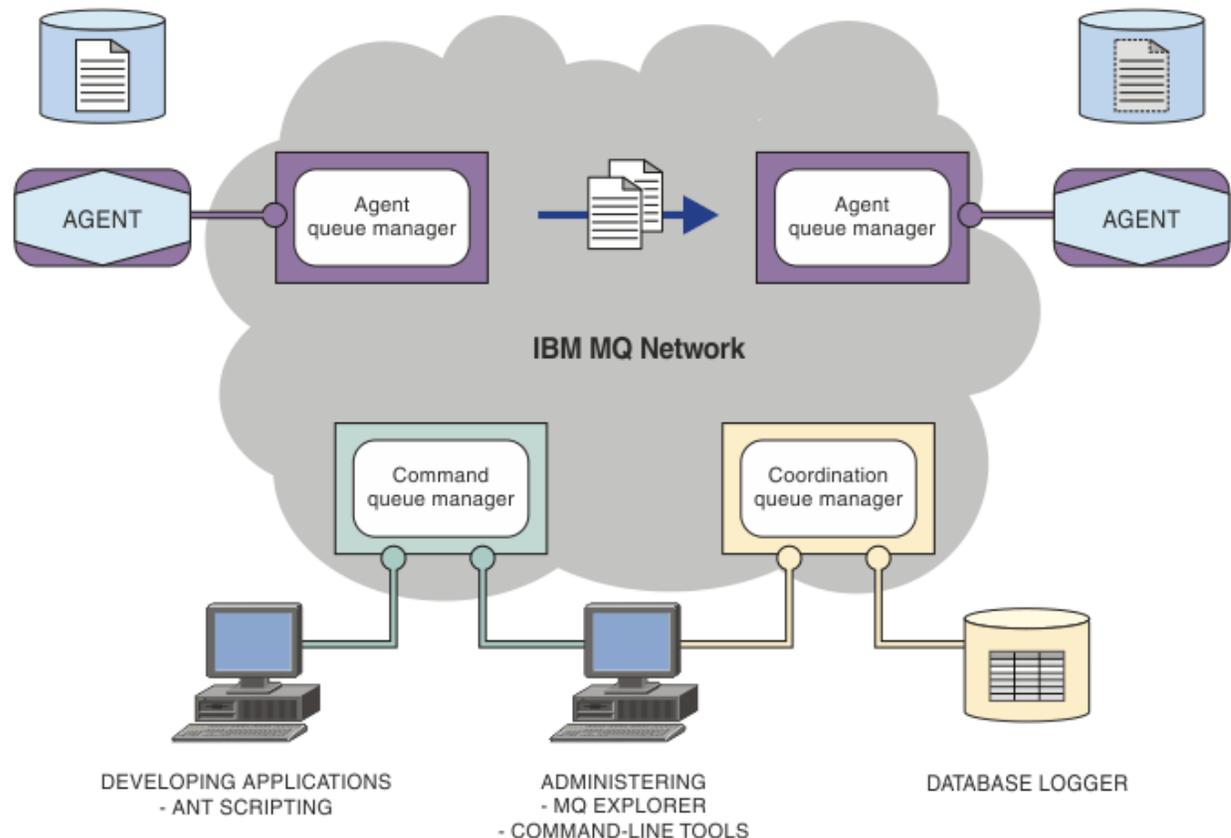
Le gestionnaire de file d'attente de coordination publie les informations d'audit et de transfert de fichier. Il représente un point unique pour la collecte des informations sur l'agent, le statut de transfert et l'audit de transfert. Il n'est pas nécessaire que le gestionnaire de file d'attente de coordination soit disponible pour que les transferts aient lieu. Si le gestionnaire de file d'attente de coordination devient temporairement indisponible, les transferts continuent normalement. Les messages de statut et d'audit sont stockés dans les gestionnaires de files d'attente des agents jusqu'à ce que le gestionnaire de file d'attente de coordination devienne disponible et puisse les traiter normalement.

Les agents s'enregistrent auprès du gestionnaire de file d'attente de coordination et publient les informations détaillées les concernant dans ce dernier. Les informations sur les agents sont utilisées par le plug-in Managed File Transfer pour permettre le démarrage des transferts depuis IBM MQ Explorer. Les

informations sur les agents collectées sur le gestionnaire de file d'attente de coordination sont également utilisées par les commandes pour afficher les informations sur les agents et le statut des agents.

Les informations sur le statut de transfert et l'audit de transfert sont publiées sur le gestionnaire de file d'attente de coordination. Les informations sur le statut de transfert et l'audit de transfert sont utilisées par le plug-in Managed File Transfer pour surveiller la progression des transferts à partir d'IBM MQ Explorer. Les informations sur l'audit de transfert stockées dans le gestionnaire de file d'attente de coordination peuvent être conservées à des fins d'audit.

Le gestionnaire de file d'attente de commandes permet de se connecter au réseau IBM MQ ; il s'agit du gestionnaire de files d'attente auquel vous vous connectez lorsque vous exécutez des commandes Managed File Transfer.



Concepts associés

«Managed File Transfer», à la page 5

Managed File Transfer transfère des fichiers entre des systèmes d'une manière gérée et auditable, quelle que soit la taille des fichiers ou des systèmes d'exploitation utilisés.

«Fonctionnement de MFT avec IBM MQ», à la page 10

Managed File Transfer interagit de diverses façons avec IBM MQ.

«MFT Présentation du scénario», à la page 29

Cette section répertorie les topologies Managed File Transfer communes ainsi qu'un scénario qui configure le système et transfère un message de test.

V 9.0.5 Présentation de MFT REST API

Présentation des améliorations apportées à REST API pour Managed File Transfer.

Depuis la IBM MQ 9.0.5, REST API ajoute la prise en charge de certaines commandes Managed File Transfer , y compris la liste des transferts, ainsi que des détails sur les agents de transfert de fichiers.

Pour plus d'informations, voir [les améliorations apportées à REST API](#).

Nouveautés et modifications apportées à MFT dans la IBM MQ 9.0

Liens vers des informations sur les principales fonctions Managed File Transfer nouvelles et modifiées dans IBM MQ 9.0.

IBM MQ 9.0.0

► V 9.0.0

- V 9.0.0 [Managed File Transfer Amélioration de la journalisation de l'agent de pont de protocole](#)
- V 9.0.0 [Retrait de la prise en charge de Managed File Transfer Web Gateway](#)

Editions IBM MQ 9.0.xContinuous Delivery

CD

- Windows ► V 9.0.1 ► Linux [Managed File Transfer délai de reprise configurable pour les transferts bloqués](#)
- Windows ► V 9.0.1 ► Linux [Package redistribuable de l'Managed File Transfer Agent](#)
- Windows ► V 9.0.1 ► Linux [Modifications apportées aux valeurs de sortie **DISPLAY CHSTATUS** pour les agents Managed File Transfer distants](#)
- Windows ► V 9.0.1 ► Linux [Modifications apportées aux transferts de messages vers des fichiers dans Managed File Transfer](#)
- V 9.0.1 [Nouvelle propriété d'agent MFT additionalWildcardSandboxChecking](#)
- Windows ► Linux ► V 9.0.2 [Amélioration de la dénomination des abonnements créés par le plug-in IBM MQ Explorer MFT](#)
- Windows ► Linux ► V 9.0.2 [Amélioration des informations sur l'agent MFT lorsque le dernier statut signalé est UNKNOWN](#)
- V 9.0.2 ► z/OS ► MQ Adv. VUE [Nouvelle propriété d'agent **adminGroup** à utiliser avec les agents MFT sous z/OS](#)
- Windows ► V 9.0.3 ► z/OS ► Linux [Journalisation des événements du moniteur de ressources Managed File Transfer](#)
- Modifications apportées à l'enregistrement de l'identificateur de produit (PID) z/OS
- V 9.0.3 ► z/OS ► MQ Adv. VUE [Connexions à distance de l'agent avec IBM MQ Advanced for z/OS, Value Unit Edition](#)
- Windows ► z/OS ► Linux [Modification du comportement de la commande Managed File Transfer **fteCleanAgent**](#)
- Windows ► Linux [Comportement corrigé pour messagePublicationFormat=mixed dans les propriétés d'installation de Managed File Transfer](#)
- Windows ► V 9.0.3 ► z/OS ► Linux [Activation de l'authentification MQCSP pour Managed File Transfer](#)
- Windows ► V 9.0.4 ► Linux ► AIX [Possibilité d'exécuter Managed File Transfer Logger en mode client](#)
- Multi ► V 9.0.5 [«Présentation de MFT REST API», à la page 11](#)

IBM MQ 9.0.0.x Long Term Support

LTS

- **V 9.0.0.1** Modifications de la commande **fteMigrateAgent**
- **V 9.0.0.1** Nouvelle propriété d'agent MFT `addCommandPathToSandbox`
- **V 9.0.0.1** Nouvelle propriété d'agent MFT `additionalWildcardSandboxChecking`
- **V 9.0.0.1** **z/OS** Nouvelle propriété d'agent `adminGroup` à utiliser avec les agents MFT sur z/OS
- **V 9.0.0.2** Modification du comportement de la commande Managed File Transfer **fteCleanAgent**
- **V 9.0.0.2** Activation de l'authentification MQCSP dans Managed File Transfer
- **V 9.0.0.2** Nouvelle propriété **messagePublicationFormat** pour le fichier Managed File Transfer `installation.properties`
- **V 9.0.0.2** Vérification de comparaison de l'identificateur et de la valeur de transfert de l'attribut `groupId` restaurée pour les transferts de message à fichier
- **V 9.0.5** Possibilité d'utiliser des listes séparées par le signe deux-points dans les commandes `fteSetAgentTraceLevel` et `fteSetLoggerTracelevel`

Information associée

V 9.0.0

Nouveautés et modifications apportées à IBM MQ 9.0.0

CD

Nouveautés et modifications dans la IBM MQ 9.0.x, Continuous Delivery

LTS

Modifications apportées dans la IBM MQ 9.0.x, Long Term Support

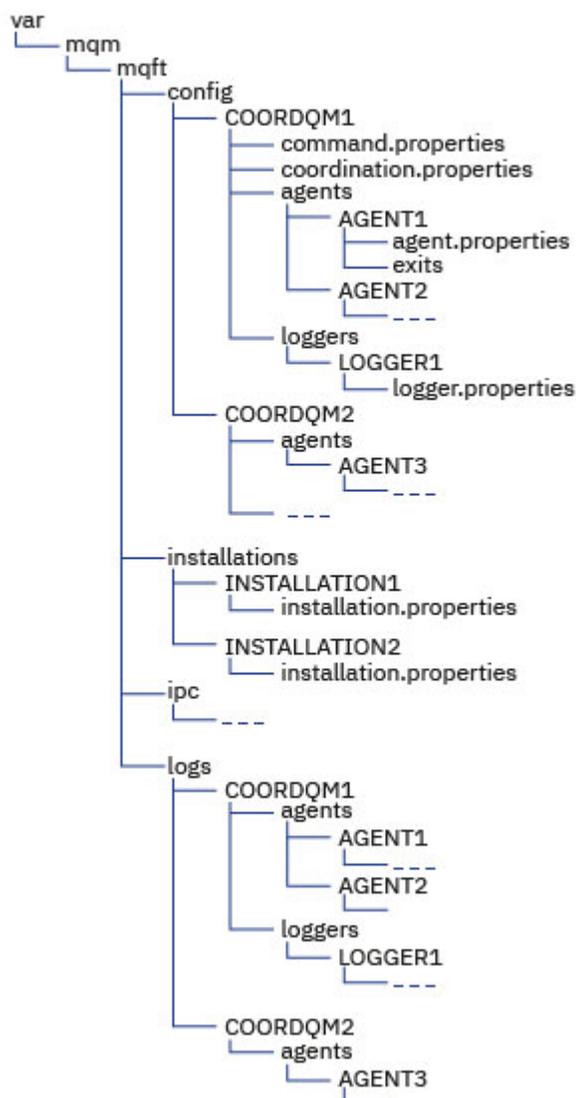
Remarques sur la migration pour MFT

Si vous prévoyez de migrer Managed File Transfer, consultez les informations suivantes.

Présentation de la configuration directement après l'installation dans MFT

Le diagramme présente la présentation de la configuration directement après l'installation, telle qu'elle est dans IBM MQ.

IBM MQ Managed File Transfer configuration



Répertoires de configuration par défaut:

Les informations sont réparties dans quatre sous-répertoires distincts: config, installations, ipc et logs.

Les répertoires racine du produit par défaut (MQ_DATA_PATH) sont les suivants:

- Systèmes UNIX : /var/mqm
- Systèmes Linux : /var/mqm
- Windows : l'emplacement du répertoire de configuration dépend de l'emplacement de votre installation principale d'IBM MQ. Les emplacements par défaut des installations principales sont les suivantes :
 - 32 bits : C:\Program Files (x86) \IBM\WebSphere MQ
 - 64 bits : C:\Program Files\IBM\MQ

Les sous-répertoires de configuration sont les suivants:

- Le répertoire `MQ_DATA_PATH/mqft/config` contient les éléments de la configuration en lecture seule pour les processus Managed File Transfer, par exemple `agent.properties` et `command.properties`.

- Le répertoire `MQ_DATA_PATH/mqft/installations` contient des informations de configuration pour chaque installation. Le contenu de ce répertoire est équivalent au contenu du fichier `wmqfte.properties`.
- Le répertoire `MQ_DATA_PATH/mqft/ipc` contient les ressources IPC utilisées en interne pour communiquer entre les composants Managed File Transfer. Applicable aux systèmes UNIX et Linux uniquement.
- Le répertoire `MQ_DATA_PATH/mqft/logs` contient les éléments de la configuration écrits par les processus Managed File Transfer. par exemple les fichiers journaux et d'information de trace.

Un fichier `installation.properties`

Linux **UNIX** Sur les systèmes UNIX and Linux, l'emplacement par défaut est `MQ_DATA_PATH/mqft/installations/installation_name`

Windows Sous Windows, l'emplacement par défaut est `MQ_DATA_PATH\mqft\installations\installation_name`

Un fichier `logger.properties`

Ce fichier intègre des informations de propriété pour les consignateurs de fichiers autonomes, les consignateurs de base de données autonomes et les consignateurs de base de données Java EE .

L'emplacement par défaut est `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

Modifications de la sécurité

Pour IBM WebSphere MQ 7.5 ou version ultérieure, seuls les utilisateurs qui sont des administrateurs (membres du groupe `mqm`) peuvent exécuter la liste de commandes **fte** suivante:

- [«fteChangeDefaultConfigurationOptions»](#), à la page 393
- [«Agent fteCreate\(créer un agent MFT \)»](#), à la page 399
- [«fteCreateBridgeAgent \(créer et configurer un agent de pont de protocole MFT \)»](#), à la page 403
- [«fteCreateCDAgent \(créer un agent de pont Connect:Direct \)»](#), à la page 413
- [«fteCreateLogger \(création d'un fichier MFT ou d'un consignateur de base de données\)»](#), à la page 418
- [«fteDeleteAgent»](#), à la page 473
- [«fteDeleteLogger»](#), à la page 476
- [«Agent fteMigrate: migration d'un agent FTE 7.0 vers MQ 7.5 ou version ultérieure»](#), à la page 494
- [«fteMigrateConfigurationOptions: migration d'une configuration FTE 7.0 vers MQ 7.5 ou version ultérieure»](#), à la page 497
- [«fteMigrateLogger: migration d'un consignateur de base de données FTE 7.0 vers MQ 7.5 ou version ultérieure»](#), à la page 499
- [«Agent fteModify\(exécutez un agent MFT en tant que service Windows \)»](#), à la page 502
- [«fteModifyLogger \(exécutez un consignateur MFT en tant que service Windows \)»](#), à la page 504
- [«Commandes fteSetup: créez le fichier MFT `command.properties`»](#), à la page 524
- [«fteSetupCoordination»](#), à la page 527

Multi Lorsque vous utilisez IBM WebSphere MQ 7.5 ou une version ultérieure sous Multiplateformes, seul l'utilisateur sur lequel s'exécute le processus d'agent peut exécuter la commande **fteSetAgentTraceLevel**.

V 9.0.2 **z/OS** Pour z/OS, pour Continuous Delivery à partir de IBM MQ 9.0.2, la commande **fteSetAgentTraceLevel** peut être exécutée par:

- Le même ID utilisateur que celui sous lequel le processus d'agent s'exécute.
- Les membres du groupe spécifiés par la propriété d'agent **adminGroup**.

V 9.0.0.1 Cela s'applique également à l'édition Long Term Support de IBM MQ 9.0.0 Fix Pack 1.

V 9.0.2 **z/OS** Pour z/OS, pour Continuous Delivery à partir de IBM MQ 9.0.2, si la commande **fteShowAgentDetails** est exécutée pour afficher des informations sur un agent local, la commande peut être exécutée par:

- Le même ID utilisateur que celui sous lequel le processus d'agent s'exécute.
- Les membres du groupe spécifiés par la propriété d'agent **adminGroup**.

V 9.0.0.1 Cela s'applique également à l'édition Long Term Support de IBM MQ 9.0.0 Fix Pack 1.

Pour plus d'informations, voir la propriété **adminGroup** dans [«Le fichier MFT agent.properties»](#), à la page 564.

Modifications apportées à la sécurité dans IBM MQ 8.0

Si vous exécutez Managed File Transfer dans IBM WebSphere MQ 7.0 et migrez vers IBM MQ 8.0, les informations d'ID utilisateur dans le fichier `MQMFTCcredentials.xml` sont transmises au gestionnaire de files d'attente, mais ne sont pas utilisées.

En effet, la transmission des informations d'ID utilisateur et de mot de passe est prise en charge dans IBM MQ 8.0 seulement.

commandPath et bacs à sable d'agent

Pour IBM MQ 8.0 et les versions ultérieures, si un agent a été configuré avec un bac à sable d'agent et que la propriété d'agent `commandPath` a été définie, les répertoires spécifiés par `commandPath` sont automatiquement ajoutés aux chemins refusés au démarrage de l'agent. Si la propriété `commandPath` est définie sur un agent qui n'est pas configuré avec un bac à sable d'agent, un nouveau bac à sable est configuré automatiquement et les répertoires spécifiés par `commandPath` sont ajoutés aux répertoires refusés au démarrage de l'agent.

Si la propriété `commandPath` est définie sur un agent qui n'est pas configuré avec un bac à sable d'agent, un nouveau bac à sable est configuré automatiquement et les répertoires spécifiés par `commandPath` sont ajoutés aux répertoires refusés au démarrage de l'agent.

Pour plus d'informations sur la propriété `commandPath`, voir [«Propriété commandPath MFT»](#), à la page 374 et [«Le fichier MFT agent.properties»](#), à la page 564.

commandPath et bacs à sable d'utilisateur

Pour IBM MQ 8.0 et les versions ultérieures, si un agent a été configuré avec un ou plusieurs bacs à sable d'utilisateur et que la propriété d'agent `commandPath` est définie, les répertoires spécifiés par `commandPath` (et tous leurs sous-répertoires) sont automatiquement ajoutés en tant qu'éléments `<exclude>` aux éléments `<read>` et `<write>` pour chaque bac à sable d'utilisateur au démarrage de l'agent.

Pour plus d'informations sur la propriété `commandPath`, voir [«Propriété commandPath MFT»](#), à la page 374 et [«Le fichier MFT agent.properties»](#), à la page 564.

Migration des agents MFT qui s'exécutent en tant que service Windows depuis IBM WebSphere MQ 7.5 vers IBM MQ 9.0

Entre IBM WebSphere MQ 7.5, IBM MQ 8.0 et IBM MQ 9.0, le chemin d'installation par défaut d'IBM MQ a changé sur la plateforme Windows.

Si un gestionnaire de files d'attente est migré depuis IBM WebSphere MQ 7.5 vers IBM MQ 8.0 ou IBM MQ 9.0, les applications qui s'exécutent sur le même système que le gestionnaire de files d'attente doivent être reconfigurées pour charger les bibliothèques IBM MQ depuis le nouvel emplacement d'installation.

Cela concerne notamment les agents Managed File Transfer IBM MQ qui s'exécutent en tant que service Windows.

Effectuez la procédure suivante :

1. Arrêtez les agents qui sont associés au gestionnaire de files d'attente d'IBM WebSphere MQ 7.5. Voir [Stopping an MFT agent](#) pour plus d'informations. Par exemple, émettez la commande suivante :

```
fteStopAgent <agent_name>
```

2. Modifiez l'agent pour retirer la définition de service de l'agent. Voir [Run an MFT agent as a Windows service](#) pour plus d'informations.

Par exemple, exécutez la commande suivante :

```
fteModifyAgent -agentName <agent_name>
```

3. Ensuite, migrez le gestionnaire de files d'attente d'agent depuis IBM WebSphere MQ 7.5 vers la version plus récente à l'aide de la commande **setmqm** . Pour plus d'informations sur la commande **setmqm** , voir [association d'un gestionnaire de files d'attente à une installation](#) .
4. Enfin, modifiez l'agent pour le reconfigurer de sorte qu'il s'exécute à nouveau en tant que service Windows avec la commande **fteModifyAgent**.

Exemple :

```
fteModifyAgent -agentName AGENT1 -s -su fteuser -sp ftepassword
```

Migration d'un agent IBM WebSphere MQ File Transfer Edition 7.0 vers IBM WebSphere MQ 7.5 ou version ultérieure

Utilisez la commande **fteMigrateAgent** pour migrer un agent IBM WebSphere MQ File Transfer Edition 7.0 vers IBM WebSphere MQ 7.5 ou version ultérieure. Si un agent est configuré pour s'exécuter en tant que service Windows, vous devez appliquer la procédure suivante facultative dans cette tâche.

Avant de commencer

V 9.0.0 Depuis IBM MQ 9.0, Managed File Transfer ne prend pas en charge les agents Web. Si vous tentez d'utiliser la commande **fteMigrateAgent** pour migrer un agent Web à partir d'une version antérieure à IBM MQ 9.0, un message d'erreur s'affiche et indique que la migration d'un agent Web n'est pas pris en charge.

Identifiez les instances de l'agent IBM WebSphere MQ File Transfer Edition installé comme partie d'IBM WebSphere MQ File Transfer Edition.

IBM i Les informations de cette rubrique ne s'appliquent pas à IBM i. Si vous utilisez IBM i, migrez votre gestionnaire de files d'attente vers IBM MQ 8.0 avant d'utiliser Managed File Transfer dans IBM MQ 8.0.

Pourquoi et quand exécuter cette tâche

Pour effectuer une migration vers IBM WebSphere MQ 7.5 ou une version ultérieure, arrêtez tout d'abord l'agent, migrez l'installation de gestionnaire de files d'attente vers IBM WebSphere MQ 7.5 ou une version ultérieure puis sélectionnez les composants File Transfer. Une fois que le gestionnaire de files d'attente a été migré, vous pouvez migrer la configuration de l'agent vers IBM WebSphere MQ 7.5 ou une version ultérieure, en utilisant la commande **fteMigrateAgent**. Démarrez l'agent migré en utilisant le gestionnaire de files d'attente IBM WebSphere MQ 7.5 ou version ultérieure.

Si l'agent se connecte à son gestionnaire de files d'attente en utilisant le mode liaisons IBM WebSphere MQ et que ce gestionnaire utilise IBM WebSphere MQ 7.0.1 Fix Pack 6 ou une version ultérieure, vous pouvez également effectuer une migration côte à côte en utilisant plusieurs installations ainsi que la

procédure présentée ci-dessous. Si vous ne souhaitez pas effectuer de migration côté à côté, migrez le gestionnaire de files d'attente et l'agent en suivant à la place les étapes décrites plus haut.

Si l'agent se connecte à son gestionnaire de files d'attente en utilisant le mode liaisons IBM WebSphere MQ et que ce gestionnaire utilise IBM WebSphere MQ 7.0.1 Fix Pack 5 ou une version antérieure, vous pouvez migrer cette installation vers IBM WebSphere MQ 7.0.1 Fix Pack 6, pour autoriser une migration côté à côté ou vous devez migrer cette installation directement vers IBM WebSphere MQ 7.5 ou une version ultérieure.

Si l'agent se connecte à son gestionnaire de files d'attente via un réseau en tant que client IBM WebSphere MQ, vous pouvez migrer l'agent vers IBM WebSphere MQ 7.5 ou une version ultérieure uniquement en suivant l'étape «7», à la page 18.

Si l'agent est configuré en tant que service Windows, vous devez exécuter la commande avec le paramètre **-f**. Pour plus d'informations, voir la section «Agent fteMigrate: migration d'un agent FTE 7.0 vers MQ 7.5 ou version ultérieure», à la page 494.

Procédure

1. Installez IBM WebSphere MQ 7.5 ou une version ultérieure parallèlement à l'installation du gestionnaire de files d'attente en sélectionnant les composants MQ Server et File Transfer.
2. Arrêtez l'agent IBM WebSphere MQ File Transfer Edition 7.0.x .
3. Déplacez le gestionnaire de files d'attente vers l'installation d'IBM WebSphere MQ 7.5 ou ultérieure.
4. Utilisez la commande **fteMigrateAgent** pour migrer la configuration d'agent de l'installation IBM WebSphere MQ File Transfer Edition 7.0.x vers la fonction d'agent intégrée à IBM WebSphere MQ 7.5 ou version ultérieure.
5. Facultatif: si l'agent IBM WebSphere MQ File Transfer Edition 7.0.x est configuré en tant que service Windows , effectuez les étapes facultatives suivantes:
 - a) Utilisez la commande **fteModifyAgent** pour reconfigurer l'agent afin qu'il ne soit plus un service Windows.
 - b) Utilisez la commande **fteModifyAgent** à partir de l'installation IBM WebSphere MQ 7.5 ou version ultérieure pour configurer la version de l'agent IBM WebSphere MQ 7.5 ou version ultérieure de telle sorte qu'il soit un service Windows.
6. Démarrez l'agent IBM WebSphere MQ 7.5 ou version ultérieure.
7. Facultatif - Pour migrer un agent se connectant en tant que client à son gestionnaire de files d'attente via un réseau, procédez comme suit :
 - a) Installez IBM WebSphere MQ 7.5 ou version ultérieure sur le système
 - b) Utilisez la commande **fteMigrateAgent** pour migrer la configuration d'agent de l'installation WMQFTE 7.0.x vers la fonction d'agent intégrée à IBM WebSphere MQ 7.5 ou version ultérieure.
 - c) Arrêtez l'agent IBM WebSphere MQ File Transfer Edition 7.0.x .
 - d) Si l'agent IBM WebSphere MQ File Transfer Edition 7.0.x est configuré en tant que service Windows , utilisez la commande **fteModifyAgent** à partir de l'installation IBM WebSphere MQ File Transfer Edition 7.0.x pour reconfigurer l'agent de sorte qu'il ne soit pas exécuté en tant que service Windows .
 - e) Si l'agent IBM WebSphere MQ File Transfer Edition 7.0.x est configuré en tant que service Windows , utilisez la commande **fteModifyAgent** à partir de l'installation IBM WebSphere MQ 7.5 ou d'une version ultérieure pour configurer l'agent IBM WebSphere MQ 7.5 ou d'une version ultérieure en tant que service Windows .
 - f) Démarrez l'agent IBM WebSphere MQ 7.5 ou version ultérieure.

Migration d'un consignateur de base de données FTE 7.0 vers MQ 7.5 ou version ultérieure

Utilisez la commande **fteMigrateLogger** pour migrer un consignateur de base de données IBM WebSphere MQ File Transfer Edition 7.0 autonome vers IBM WebSphere MQ 7.5 ou une version ultérieure.

Si vous avez configuré le consignateur de base de données en tant que service Windows, vous devez effectuer des étapes de migration supplémentaires.

Avant de commencer

Identifiez les instances du consignateur de base de données autonome. Vous devez les adapter avant les autres parties du réseau Managed File Transfer pour qu'elles puissent traiter correctement les versions les plus récentes des messages du journal de transfert qu'elles reçoivent.

Pourquoi et quand exécuter cette tâche

Pour migrer directement vers IBM WebSphere MQ 7.5 ou ultérieure, vous devez d'abord arrêter le consignateur, puis migrer cette installation vers la version de votre choix. Une fois cette migration terminée, utilisez la commande **fteMigrateLogger** pour migrer la configuration du consignateur de base de données vers la version requise de IBM MQ.

Si le gestionnaire de files d'attente est IBM WebSphere MQ File Transfer Edition 7.0.1 Fix Pack 6 ou version ultérieure, vous pouvez aussi effectuer une migration côte à côte en utilisant plusieurs installations, comme décrit en détail dans les étapes ci-après. Si vous ne voulez pas effectuer de migration côte à côte, migrez le gestionnaire de files d'attente et le consignateur en suivant les étapes décrites plus haut à la place.

Si le consignateur de base de données est connecté à un gestionnaire de files d'attente IBM WebSphere MQ File Transfer Edition 7.0.1 Fix Pack 5 ou version ultérieure, vous devez d'abord migrer cette installation vers IBM WebSphere MQ File Transfer Edition 7.0.1 Fix Pack 6, pour permettre une migration côte à côte, ou migrer cette installation directement vers IBM WebSphere MQ 7.5 ou ultérieure, puis sélectionner les composants File Transfer.

Si le consignateur de base de données est configuré en tant que service Windows, vous devez exécuter la commande **fteMigrateLogger** avec le paramètre **-f**. Pour plus d'informations, voir la section [«fteMigrateLogger: migration d'un consignateur de base de données FTE 7.0 vers MQ 7.5 ou version ultérieure»](#), à la page 499.

Procédure

1. Installez IBM WebSphere MQ 7.5 ou ultérieure parallèlement à l'installation de IBM WebSphere MQ File Transfer Edition 7.0 existante, en sélectionnant les composants MQ Server et Managed File Transfer Service.
2. Arrêtez le consignateur de base de données IBM WebSphere MQ File Transfer Edition 7.0.
3. Déplacez le gestionnaire de files d'attente vers l'installation d'IBM WebSphere MQ 7.5 ou ultérieure.
4. Utilisez la commande **fteMigrateLogger** pour migrer la configuration du consignateur de base de données de l'installation IBM WebSphere MQ File Transfer Edition 7.0 vers le consignateur de base de données IBM WebSphere MQ 7.5 ou version ultérieure.
5. Créez les tables nécessaires pour démarrer le gestionnaire de journalisation de la base de données en exécutant les scripts SQL dans `MQ_INSTALLATION_PATH/mqft/sql`. Il existe un script pour chaque incrément du niveau de produit ; exécutez tous ceux qui sont appropriés dans l'ordre.

Vous devez exécuter les scripts SQL de mise à niveau dans l'ordre des versions, en commençant par leur niveau en cours d'IBM WebSphere MQ File Transfer Edition ou de Managed File Transfer. Les scripts disponibles, où ******* peut correspondre à Db2, Oracle ou z/OS, sont les suivants :

- `ftelog_tables_***_701-702.sql`
- `ftelog_tables_***_702-703.sql`
- `ftelog_tables_***_703-704.sql`
- `ftelog_tables_***_704-750.sql`
- `ftelog_tables_***_750-7502.sql`
- `ftelog_tables_***_7502-800.sql`

Par exemple, si vous utilisez IBM WebSphere MQ File Transfer Edition 7.0.3 et migrez vers IBM MQ 8.0.0.0, exécutez 703-704, 704-750, 750-7502 et 7502-800.

Remarque :  Sous z/OS, vous pouvez passer directement de 704 à 800, à l'aide de `ftelog_tables_zos_704-800.sql`

Les scripts jusqu'à `ftelog_tables_***_704-750.sql` sont inclus dans IBM MQ 8.0.0.0 et se trouvent dans le répertoire `MQ_INSTALL ATION_PATH/mqft/sql`. Les mises à niveau au-delà de 750 n'ont pas été livrées avec IBM MQ 8.0.0.0 ; si elles ne se trouvent pas dans le répertoire, vous pouvez les télécharger depuis le lien spécifié dans l'[APAR IT01841](#).

6. Facultatif : si le consignateur de base de données IBM WebSphere MQ File Transfer Edition 7.0 a été configuré en tant que service Windows, procédez comme suit :
 - a) Reconfigurez le consignateur de base de données pour qu'il ne s'agisse plus d'un service Windows avec la commande `fteModifyDatabaseLogger`.
 - b) Reconfigurez le Managed File Transfer dans le consignateur IBM WebSphere MQ 7.5 ou version ultérieure pour qu'il s'agisse d'un service Windows à l'aide de la commande `fteModifyLogger`.
7. Démarrez le Managed File Transfer dans le consignateur de base de données IBM WebSphere MQ 7.5 ou version ultérieure.

Résultats

Le consignateur de base de données a été migré de IBM WebSphere MQ File Transfer Edition 7.0 vers IBM WebSphere MQ 7.5 ou ultérieure.

Augmentation de la taille de page de la base de données de journaux sous Db2 sous Windows, UNIX ou Linux

Si votre base de données est Db2 sur un système Windows, UNIX ou Linux et que vous avez créé votre base de données de journaux avec une taille de page inférieure à 8 Ko, vous devez augmenter la taille de page de la base de données avant de migrer vers les tables IBM WebSphere MQ File Transfer Edition 7.0.3 ou versions ultérieures.

Procédure

1. Si vous n'avez pas déjà arrêté votre consignateur de base de données, arrêtez-le avec la commande `fteStopDatabaseLogger`.
2. Effectuez une sauvegarde de votre base de données de journalisation à l'aide des outils fournis par Db2.
3. Utilisez la commande Db2 **export** pour transférer les données depuis vos tables de base de données de journalisation vers des fichiers sur le disque.

Remarque : vous devez spécifier des fichiers d'objets LOB pour les tables qui incluent des objets LOB. Ces tables s'appellent `CALL_RESULT` et `METADATA`.

4. Supprimez vos tables de base de données de journalisation.
5. Créez un espace table avec une taille de page d'au moins 8 ko et un pool de mémoire tampon associé dont la taille de page est d'au moins 8 ko.
Attribuez un nom à votre nouvel espace table. Exemple : `FTE8KSPACE`.
6. Editez le fichier `ftelog_tables_db2.sql` pour que les commandes créent des tables dans le nouvel espace table.
Dans le fichier `ftelog_tables_db2.sql`, remplacez toutes les occurrences du texte `IN "USERSPACE1"` par `IN "new_tablespace_name"`. Par exemple, remplacez `IN "USERSPACE1"` par `IN "FTE8KSPACE"`.
7. Exécutez les commandes SQL dans le fichier `ftelog_tables_db2.sql` pour votre base de données.
8. Utilisez la commande Db2 **load** pour transférer les données exportées dans les nouvelles tables.

Remarque :

- **Mappez les noms de colonne en fonction des noms de colonne figurant dans le fichier d'entrée.** Assurez-vous que les noms de colonne d'entrée et les noms de colonne cible correspondent dans les tables dont la structure a changé.
 - Vous devez spécifier le comportement IDENTITY OVERRIDE dans la colonne d'identité de toutes les tables, sauf MONITOR et TRANSFER. La spécification de ce comportement garantit que les ID de ligne ne sont pas régénérés au cours de l'opération de chargement.
9. Exécutez la commande Db2 **set integrity** avec les valeurs de statut d'intégrité **immediate** et **checked** pour les tables suivantes, dans l'ordre indiqué :
- CALL_ARGUMENT
 - MONITOR
 - MONITOR_ACTION
 - MONITOR_EXIT_RESULT
 - MONITOR_METADATA
 - SCHEDULE_ACTION
 - SCHEDULE
 - SCHEDULE_ITEM
 - TRANSFER
 - TRANSFER_CALLS
 - TRANSFER_EVENT
 - TRANSFER_ITEM
 - TRANSFER_STATS
 - TRIGGER_CONDITION
10. Dans les tables comportant des colonnes d'ID générés, définissez les générateurs d'ID pour qu'ils commencent par une valeur d'un point supérieur à la valeur d'ID la plus élevée existante.
- Les tables suivantes contiennent des colonnes d'ID générés :
- AUTH_EVENT
 - CALL
 - CALL_ARGUMENT
 - CALL_RESULT
 - FILE_SPACE_ENTRY
 - METADATA
 - MONITOR_ACTION
 - MONITOR_EXIT_RESULT
 - MONITOR_METADATA
 - SCHEDULE
 - SCHEDULE_ACTION
 - SCHEDULE_ITEM
 - SCHEDULE_SPEC
 - TRANSFER_CALLS
 - TRANSFER_CD_NODE
 - TRANSFER_CORRELATOR
 - TRANSFER_EVENT
 - TRANSFER_EXIT
 - TRANSFER_ITEM

- TRANSFER_ITEM_ATTRIBUTES
- TRANSFER_STATS
- TRIGGER_CONDITION

Pour associer les ID générés de ces tables à la valeur correcte, effectuez les étapes suivantes pour chaque table :

a) Déterminez la valeur d'ID maximale dans les données existantes.

Vous pouvez obtenir cette valeur en exécutant l'instruction SQL suivante :

```
SELECT MAX(ID) FROM FTELOG.table_name
```

La valeur renvoyée depuis cette commande est l'ID existant maximal dans la table spécifiée.

b) Modifiez la table afin de définir le générateur d'ID de sorte qu'il commence à partir d'une nouvelle valeur supérieure d'un point à la valeur renvoyée par l'étape précédente.

Vous pouvez définir cette valeur en exécutant l'instruction SQL suivante :

```
ALTER TABLE FTELOG.table_name ALTER COLUMN ID RESTART WITH value
```

Tâches associées

«Migration du consignateur de base de données autonome vers le consignateur de base de données Java EE pour MFT», à la page 145

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 Java EE . 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 JEE. Sauvegardez votre base de données avant la migration.

«Migration des tables de base de données sous Db2 sous z/OS vers IBM MQ 8.0», à la page 22

Si votre base de données est Db2 sur un système z/OS , vous devez effectuer les étapes suivantes pour migrer de IBM WebSphere MQ File Transfer Edition 7.0.3 vers IBM WebSphere MQ File Transfer Edition 7.0.4 et de IBM WebSphere MQ File Transfer Edition 7.0.4 vers Managed File Transfer dans IBM MQ 8.0. Les tables Db2 possèdent différentes structures provenant des éditions précédentes. Par exemple, certaines tables comportent de nouvelles colonnes, et certaines colonnes de caractères variables peuvent être plus grandes ; ainsi, les tables des éditions précédentes doivent être migrées vers le format Db2.

Migration des tables de base de données sous Db2 sous z/OS vers IBM MQ 8.0

Si votre base de données est Db2 sur un système z/OS , vous devez effectuer les étapes suivantes pour migrer de IBM WebSphere MQ File Transfer Edition 7.0.3 vers IBM WebSphere MQ File Transfer Edition 7.0.4 et de IBM WebSphere MQ File Transfer Edition 7.0.4 vers Managed File Transfer dans IBM MQ 8.0. Les tables Db2 possèdent différentes structures provenant des éditions précédentes. Par exemple, certaines tables comportent de nouvelles colonnes, et certaines colonnes de caractères variables peuvent être plus grandes ; ainsi, les tables des éditions précédentes doivent être migrées vers le format Db2.

Pourquoi et quand exécuter cette tâche

Managed File Transfer dans IBM MQ 8.0 prend en charge le transfert de fichiers très volumineux, dont la taille est supérieure à celle pouvant être stockée dans un nombre entier (32 bits). Il existe une définition qui utilise des nombres de type BIGINT (64 bits). Vous pouvez choisir d'utiliser des définitions de type INTEGER, appelées `ftelog_tables_zos.sql`, ou des définitions de type BIGINT, appelées `ftelog_tables_zos_bigint.sql`, qui se trouvent dans le répertoire `prod/mqf/sql`.

Le type de données BIGINT est disponible dans IBM WebSphere MQ File Transfer Edition 7.0.4 Fix Pack 3 ou Managed File Transfer dans IBM WebSphere MQ 7.5.0 Fix Pack 1 ou version ultérieure.

Pour activer l'utilisation des types de données BIGINT, vous devez utiliser Db2 9.1 ou une version ultérieure. Les types de données INTEGER sont utilisés pour les zones qui indiquent les tailles des fichiers qui sont transférés et l'ID de table associé à chaque transfert. Si vous voulez journaliser les transferts dont les tailles de fichier sont supérieures à 2 Go ou si vous voulez stocker plus de 2 147 483 648 transferts individuels dans votre base de données, vous devez utiliser le fichier SQL BIGINT.

La liste suivante présente les processus à suivre :

1. Vous disposez de tables existantes provenant de IBM WebSphere MQ 7.0. Elles possèdent un schéma, par exemple FTELOG.
2. Créez des tables de IBM MQ 8.0 à l'aide d'un nom de schéma différent, par exemple FTEV8. Cela vous permet de copier des données de FTELOG . table vers FTE8 . table.
3. Copiez les données dans la nouvelle table.
4. Définissez les valeurs d'ID générées dans les nouvelles tables.
5. Exécutez la commande **fteMigrateLogger** pour déplacer le fichier de propriétés dans un nouvel emplacement dans la structure de répertoire.
6. Editez le fichier de propriétés du consignateur afin de spécifier le nouveau schéma (FTEV8).
7. Editez le JCL du consignateur existant afin qu'il utilise les bibliothèques de la IBM MQ 8.0 de Managed File Transfer.
8. Démarrez le consignateur.
9. Une fois que le consignateur fonctionne, vous pouvez supprimer les tables FTELOG.

Dans la description ci-dessous, le produit Managed File Transfer est installé dans le répertoire /HMF8800 sur les systèmes USS.

Procédure

1. Si vous n'avez pas déjà arrêté votre consignateur de base de données, arrêtez-le avec la commande **fteStopDatabaseLogger** sur les systèmes USS ou **P loggerjob**.
2. Emettez la commande **ls /HMF8800/mqft/sql** pour répertorier les fichiers SQL dans le répertoire. Si vous prévoyez d'utiliser des nombres de type BIGINT, copiez `ftelog_tables_zos_bigint.sql` dans votre répertoire de base ; sinon, copiez `ftelog_tables_zos.sql`.
3. Editez le fichier que vous avez déplacé dans votre répertoire de base :
 - a) Remplacez `ftelog` par le nom de schéma pour les nouvelles tables.
 - b) Assurez-vous que chaque index porte un nom unique.
Pour ce faire, dans une session d'édition :
 - i) Excluez toutes les lignes.
 - ii) Recherchez 'CREATE UNIQUE INDEX ' ALL.
 - iii) Changez `_KEY _K8Y ALL NX`.
4. Vérifiez le fichier pour vous assurer que toutes les instructions figurent dans la colonne 71. Si tel n'est pas le cas, divisez la ligne avant la colonne 71.
5. Il est possible d'utiliser ce fichier comme entrée pour SQL ; ou bien, vous pouvez le copier dans un ensemble de données partitionnées. Pour ce faire, éditez cet ensemble de données partitionnées et utilisez la commande **copy** en spécifiant le répertoire et le nom de fichier.
6. Vérifiez les définitions avec votre administrateur Db2, car il peut exister des normes à respecter pour les sites.
7. Effectuez la procédure suivante :
 - a) Copiez le fichier `_zos_704-800.sql` situé dans le répertoire `/HMF8800/mqft/sql/ftelog_tables` dans votre répertoire de base.

- b) Editez ce fichier. Remplacez FTESRC par votre schéma existant (FTELOG) et FTEDEST par le nouveau schéma (FTEV8).
 - c) Vérifiez le fichier pour vous assurer que toutes les instructions figurent dans la colonne 71. Si tel n'est pas le cas, divisez la ligne avant la colonne 71.
 - d) Si vous disposez de **DB2 RUNSTATS** jcl pour les tables Managed File Transfer , créez un nouveau travail spécifiant le nouveau schéma et les nouvelles tables.
8. Certaines tables possèdent un ID généré en vue de l'application d'un identificateur unique à chaque ligne et vous devez définir ces identificateurs.

Les tables suivantes contiennent des colonnes d'ID générés :

- AUTH_EVENT
- CALL
- CALL_ARGUMENT
- CALL_RESULT
- FILE_SPACE_ENTRY
- METADATA
- MONITOR_ACTION
- MONITOR_EXIT_RESULT
- MONITOR_METADATA
- SCHEDULE
- SCHEDULE_ACTION
- SCHEDULE_ITEM
- SCHEDULE_SPEC
- TRANSFER_CALLS
- TRANSFER_CD_NODE
- TRANSFER_CORRELATOR
- TRANSFER_EVENT
- TRANSFER_EXIT
- TRANSFER_ITEM
- TRANSFER_ITEM_ATTRIBUTES
- TRANSFER_STATS
- TRIGGER_CONDITION

Pour associer les ID générés de ces tables à la valeur correcte, effectuez les étapes suivantes pour chaque table :

- a) Déterminez la valeur d'ID maximale dans les données existantes.

Vous pouvez obtenir cette valeur en exécutant l'instruction SQL suivante :

```
SELECT MAX(ID) FROM schema_name.table_name
```

La valeur renvoyée depuis cette commande est l'ID existant maximal dans la table spécifiée.

- b) Modifiez la table afin de définir le générateur d'ID de sorte qu'il commence à partir d'une nouvelle valeur supérieure d'un point à la valeur renvoyée par l'étape précédente.

Vous pouvez définir cette valeur en exécutant l'instruction SQL suivante :

```
ALTER TABLE schema_name.table_name ALTER COLUMN ID RESTART WITH value
```

9. Editez le fichier de propriétés de la base de données pour spécifier le nouveau nom de schéma :

- a) Si votre répertoire de configuration Managed File Transfer est /u/userid/fteconfig, vous pouvez utiliser la commande USS **find /u/userid/fteconfig -name databaselogger.properties** pour localiser le fichier de propriétés du consignateur.
 - b) Editez ce fichier et remplacez `wmqfte.database.schema` par la nouvelle valeur de schéma.
10. Exécutez les commandes suivantes pour convertir la structure de l'arborescence de répertoires au format 8.0.0 avant de tenter d'utiliser le consignateur:
- a) **fteMigrateConfigurationOptions**
 - b) **fteMigrateLogger**
- Cette commande copie `databaselogger.properties` dans `logger.properties`.
11. Editez le JCL du consignateur existant pour utiliser les bibliothèques 8.0.0 Managed File Transfer .
12. Démarrez le consignateur.
- Une fois que le consignateur fonctionne, vous pouvez supprimer les tables FTELOG de la version 7.

Migration des agents Managed File Transfer for z/OS vers une version ultérieure

Suivez les instructions ci-dessous pour migrer un agent IBM MQ Managed File Transfer for z/OS vers une version ultérieure. Cette rubrique utilise la IBM MQ 8.0 comme version initiale et la IBM MQ 9.0 comme version ultérieure.

Important : Ces instructions supposent que :

- Le gestionnaire de file d'attente de coordination et le gestionnaire de file d'attente de commandes pour la topologie Managed File Transfer for z/OS (MFT) ne changent pas au cours de la migration.
- Le gestionnaire de file d'attente d'agent n'est pas migré. Le niveau d'IBM MQ for z/OS qui existait avant la migration est conservé.
- L'agent migré est administré à l'aide de JCL.

Etape 1 - Configuration initiale

1. Installez IBM MQ Managed File Transfer for z/OS 9.0 sur la même partition logique que IBM MQ Managed File Transfer for z/OS 8.0.
2. Localisez le fichier PDSE SBFGCMDS pour IBM MQ MFT dans IBM MQ 9.0.
Pour plus d'informations sur l'ensemble de données partitionnées étendu SBFGCMDS, voir [The SBFGCMDS library](#).
3. Dans cet ensemble de données, éditez le membre BFGCOPY pour spécifier le nom de l'ensemble de données partitionnées qui doit être utilisé pour administrer l'agent.
4. Soumettez le membre BFGCOPY. Ainsi, vous copiez l'ensemble de données partitionnées étendu SBFGCMDS existant dans un nouvel ensemble de données partitionnées étendu avec le nom spécifié à l'étape «3», à la page 25.

Etape 2 - Modification du nouvel ensemble de données partitionnées étendu

1. Dans le nouvel ensemble de données partitionnées étendu que vous avez créé à l'aide du membre BFGCOPY, éditez BFGCUSTOM et mettez à jour les variables suivantes :

Variable	Valeur
++library++	Remplacez cette variable par le nom du nouvel ensemble de données partitionnées étendu.
armELEMENT	Définissez pour cette variable la même valeur que celle qui est utilisée par l'agent de la IBM MQ 8.0.

Variable	Valeur
armELEMTYPE	Définissez pour cette variable la même valeur que celle qui est utilisée par l'agent de la IBM MQ 8.0.
BFG_DATA	Définissez pour cette variable la même valeur que celle qui est utilisée par l'agent de la IBM MQ 8.0.
BFG_GROUP_NAME	Définissez pour cette variable la même valeur que celle qui est utilisée par l'agent de la IBM MQ 8.0.
BFG_JAVA_HOME	Pour cette variable, vous devez définir le chemin UNIX System Services (USS) de l'environnement d'exécution Java qui doit être utilisé pour exécuter l'agent. Définissez la même valeur que celle qui est utilisée pour l'agent de la IBM MQ 8.0.
BFG_JVM_PROPERTIES	Définissez pour cette variable la même valeur que celle qui est utilisée par l'agent de la IBM MQ 8.0.
BFG_PROD	Cette variable doit être définie sur le chemin USS pour Managed File Transfer for z/OS dans l'installation IBM MQ 9.0 .
BFG_WTO	Définissez pour cette variable la même valeur que celle qui est utilisée par l'agent de la IBM MQ 8.0.
CLEAN_AGENT_PROPS	Laissez la valeur de cette variable vierge.
connectionQMgr	Définissez pour cette variable la même valeur que celle qui est utilisée par l'agent de la IBM MQ 8.0.
coordinationQMgr	Définissez pour cette variable la même valeur que celle qui est utilisée par l'agent de la IBM MQ 8.0.
CREDENTIALS_PATH	Laissez la valeur de cette variable vierge.
DB_PROPS_PATH	Définissez pour cette variable la même valeur que celle qui est utilisée par l'agent de la IBM MQ 8.0.
DLQ_HLQ	Définissez pour cette variable la même valeur que celle qui est utilisée par l'agent de la IBM MQ 8.0.
FTE_CONFIG	Laissez la valeur de cette variable vierge.
MQ_HLQ	Définissez pour cette variable la même valeur que celle qui est utilisée par l'agent de la IBM MQ 8.0.
MQ_PATH	Définissez pour cette variable le chemin USS de l'installation d'IBM MQ associée au gestionnaire de file d'attente d'agent.
NOM	Définissez pour cette variable le nom de l'agent qui est migré.
QM	Définissez pour cette variable la même valeur que celle qui est utilisée par l'agent de la IBM MQ 8.0. Ainsi, vous garantissez que lorsque l'agent exécute la IBM MQ 9.0, il utilise le même gestionnaire de file d'attente d'agent que celui qu'il utilisait lorsqu'il exécutait la IBM MQ 8.0.
SERVICE_TYPE	AGENT
TMPDIR	Définissez pour cette variable la même valeur que celle qui est utilisée par l'agent de la IBM MQ 8.0.

2. Sauvegardez le membre BFGCUSTM modifié.
3. Soumettez BFGCUSTM. Le membre crée de nouveaux membres dans l'ensemble de données partitionnées étendu pouvant être utilisé pour administrer l'agent lorsque celui-ci utilise la IBM MQ 9.0.



Avertissement : Le membre requiert un accès exclusif à l'ensemble de données partitionnées étendu ; par conséquent, vous devez arrêter de l'utiliser lorsque BFGCUSTM est en cours d'exécution.

Etape 3 - Exécution de la migration

1. Localisez l'ensemble de données partitionnées étendu de la IBM MQ 8.0 qui est utilisé pour administrer l'agent.
2. Dans cet ensemble de données partitionnées étendu, soumettez le membre BFGAGSP pour arrêter l'agent. L'agent s'arrête une fois qu'il a terminé tous ses transferts gérés.
3. Ensuite, localisez le nouvel ensemble de données partitionnées étendu de la IBM MQ 9.0 que vous venez de créer pour l'agent.
4. Dans cet ensemble de données partitionnées étendu, soumettez le membre BFGAGST. Cette opération démarre l'agent à l'aide de IBM MQ Managed File Transfer for z/OS 9.0

Concepts associés

Utilisation de Managed File Transfer for z/OS

Vous pouvez appliquer les instructions de cette rubrique comme méthode alternative d'utilisation de Managed File Transfer dans votre entreprise, sur votre système IBM MQ for z/OS .

Migration de MFT vers une nouvelle machine exécutant un autre système d'exploitation

Etapes de base requises pour réussir la migration des configurations MFT vers un nouveau système ou une nouvelle plateforme. La tâche traite principalement de la migration de la configuration MFT, mais également de la migration du gestionnaire de files d'attente le cas échéant.

Avant de commencer

Assurez-vous que les agents que vous voulez migrer ont terminé les transferts en cours ou en attente, ainsi que d'avoir effectué une sauvegarde :

- Du gestionnaire de file d'attente de coordination
- Des gestionnaires de file d'attente d'agent
- Des agents
- Des moniteurs de ressources
- Des modèles de transfert
- Des transferts planifiés

Important : Il est peu probable que les noms d'installation d'IBM MQ sur un système correspondent aux noms d'installation sur le nouveau système, sauf si l'ancien système et le nouveau système ne comportent qu'une seule installation ou si vous spécifiez un nom d'installation au cours du processus d'installation d'IBM MQ.

Pourquoi et quand exécuter cette tâche

La procédure de migration ci-dessous repose sur le scénario où QMA est à la fois le gestionnaire de file d'attente de coordination pour la topologie et le gestionnaire de file d'attente d'agent pour un agent appelé Agent1.

Agent1 dispose d'un moniteur, d'un modèle de transfert et d'un transfert planifié. QMA se connecte également à un gestionnaire de files d'attente appelé QMB qui s'exécute sur un autre système à l'aide de ses canaux émetteur et récepteur pour les transferts de fichiers.

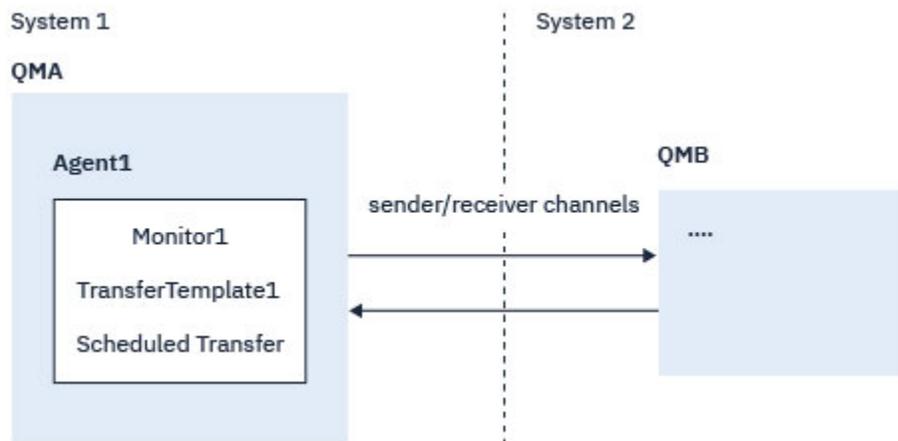


Figure 1. Migration de la configuration de MFT sur le système 1



Avertissement : La procédure ci-après explique uniquement comment sauvegarder et restaurer de configurations de MFT. Si vous migrez MFT vers une nouvelle machine exécutant le même système d'exploitation, vous pouvez sauvegarder et restaurer les données de gestionnaire de files d'attente et les fichiers journaux en copiant tous les fichiers de données de l'ancien système dans les répertoires appropriés sur le nouveau système.

Toutefois, si la nouvelle machine exécute un système d'exploitation différent, il n'est pas possible de migrer les fichiers de données ceux-ci sont créés pour une plateforme spécifique.

Procédure

1. Procédure de sauvegarde

- a) Sauvegardez la configuration du gestionnaire de files d'attente à l'aide de la commande **dmpmqcfig** pour la régénérer ultérieurement à partir de sa définition.

Exemple :

```
dmpmqcfig -m QMA -a > /mq/backups/QMA.mqsc
```

- b) Sauvegardez les fichiers de configuration de l'agent qui sont stockés dans le répertoire de données IBM MQ `/MQ_DATA_PATH/mqft`

Normalement, le répertoire `mqft` possède trois sous-répertoires : `config`, `installation` et `logs`. Ceux-ci contiennent les fichiers de données d'installation de l'agent, de configuration et de consignateur de base de données respectivement.

Si l'agent est un agent de pont de protocole, le fichier `ProtocolBridgeCredentials.xml` dans le répertoire de configuration de l'agent doit également être sauvegardé. Ce fichier définit les noms d'utilisateur et les données d'identification que l'agent de pont de protocole utilise pour s'autoriser lui-même auprès du serveur de protocole.

- c) Exportez la configuration du moniteur de ressources dans un fichier XML avec la commande MFT **ftelistMonitors**, avec l'option **-ox**.

Exemple :

```
ftelistMonitors -ma Agent1 -mn Monitor -ox Monitor1Definition.xml
```

- d) Exportez les modèles de transfert dans des fichiers XML avec la commande MFT **ftelistTemplates**, avec les options **-x** et **-o**.

Par exemple, la commande suivante crée `TransferTemplate1.xml` dans le répertoire en cours :

```
ftelistTemplates -x -o . TransferTemplate1
```

- e) Effectuez manuellement une sauvegarde des définitions de transfert planifié.

Il n'est pas possible d'exporter les définitions dans des fichiers XML, mais vous pouvez répertorier les transferts planifiés avec la commande MFT [fteListScheduledTransfers](#) et effectuer manuellement une sauvegarde des définitions.

2. Procédure de recréation

- a) Recréer le gestionnaire de files d'attente QMA après avoir installé IBM MQ et MFT sur le nouveau système.
- b) Restaurer la configuration QMA en exécutant la commande **runmqsc** à analyser dans la configuration du gestionnaire de files d'attente sauvegardée à l'étape «1.a», à la page 28

Exemple :

```
runmqsc QMA< /mq/backups/QMA.mqsc
```

- c) Recréez les canaux émetteur et récepteur qui se connectent à QMB sur le deuxième système.
- d) Dans le gestionnaire de files d'attente QMB, mettez à jour les détails de connexion, comme le nom d'hôte et le numéro de port du canal émetteur qui se connecte à QMA.
- e) Recréez Agent1 en copiant tous les fichiers de configuration de l'agent sauvegardés sur le nouveau système et démarrez l'agent.
- f) Importez le fichier XML pour Monitor1 avec la commande MFT [fteCreateMonitor](#), avec les options **-ix** et **-f**.

Exemple :

```
fteCreateMonitor -ix Monitor1Definition.xml -f
```

- g) Publiez un message comportant le contenu du fichier TransferTemplate1.xml dans le corps du message dans la rubrique SYSTEM.FTE du gestionnaire de file d'attente de coordination.

Utilisez une application autonome et spécifiez la chaîne de rubrique suivante :

```
SYSTEM.FTE/Templates/<template_id>
```

Où <template_id> est l'ID de modèle de transfert qui se trouve dans le fichier TransferTemplate1.xml.

Par exemple, si le fichier xml contient :

```
<?xml version="1.0" encoding="UTF-8"?><transferTemplateid="a7838085-0f2a-4980-b958-2dbbdfb22702" version="6.00">
```

, la chaîne de rubrique doit être :

```
SYSTEM.FTE/Templates/a7838085-0f2a-4980-b958-2dbbdfb22702
```

- h) Recréez manuellement les transferts planifiés avec la commande MFT [fteCreateTransfers](#).

MFT Présentation du scénario

Cette section répertorie les topologies Managed File Transfer communes ainsi qu'un scénario qui configure le système et transfère un message de test.

- [Topologies courantes](#)
- [Configuration du serveur de base](#)

Topologies courantes de MFT

Cette section répertorie les topologies Managed File Transfer communes. Les flèches à deux sens dans les diagrammes représentent des connexions au gestionnaire de files d'attente.

Voir «Remarques relatives à la connectivité», à la page 33 pour plus d'informations sur les options de connexion à un gestionnaire de files d'attente.

Topologie de base comportant un gestionnaire de files d'attente

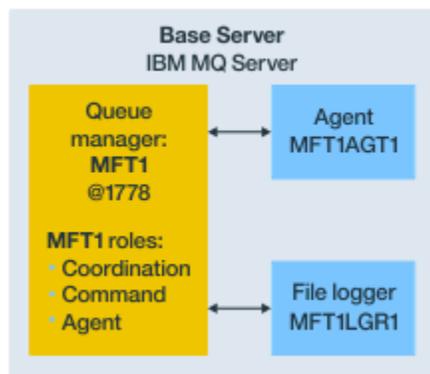


Figure 2. Topologie de base comportant un gestionnaire de files d'attente

Une topologie de base représente une configuration complète qui inclut le gestionnaire de file d'attente de coordination. Le nom de la configuration est identique au nom du gestionnaire de file d'attente de coordination. Si le nom du gestionnaire de file d'attente de coordination est MFT1, le nom de la configuration est MFT1.

La topologie de base est la première configuration Managed File Transfer que vous effectuez. Une fois la configuration de base effectuée, des agents partenaires provenant de serveurs distants sont ajoutés à la configuration de base en vue de l'échange de fichiers.

La topologie de base n'échange pas de fichiers hors du serveur de topologie de base. Toutefois, elle vous permet de déplacer des fichiers à d'autres emplacements sur le même serveur et peut être utilisée à des fins de développement.

Topologie de base comportant un agent partenaire

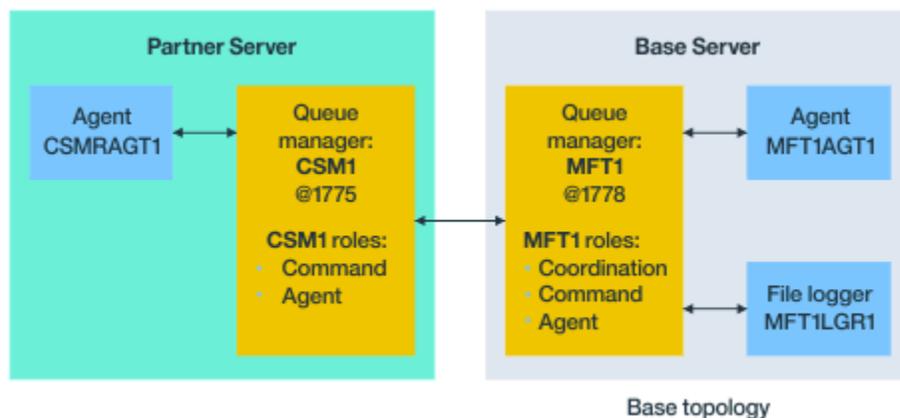


Figure 3. Topologie de base comportant un agent partenaire

Cette topologie peut échanger des fichiers entre les deux agents. Des agents partenaires supplémentaires peuvent être ajoutés de la même façon que le premier agent.

Vous pouvez utiliser un gestionnaire de files d'attente unique pour les trois rôles de gestionnaire de files d'attente Managed File Transfer, ou utiliser des gestionnaires de files d'attente dédiés pour des rôles spécifiques.

Par exemple, vous pouvez dédier un gestionnaire de files d'attente au rôle de gestionnaire de file d'attente de coordination, et les rôles de gestionnaire de file d'attente de commandes et de gestionnaire de file d'attente d'agent peuvent partager un deuxième gestionnaire de files d'attente.

La connexion entre un gestionnaire de file d'attente d'agent distant sur un serveur distinct de la configuration de base et le gestionnaire de file d'attente de coordination de la configuration de base doit être configurée en tant que client IBM MQ ou canal MQI.

La connexion au gestionnaire de file d'attente de coordination est établie par la commande **fteSetupCoordination**. Si la connexion au gestionnaire de file d'attente de coordination n'est pas configurée en tant que canal client IBM MQ sur le serveur du partenaire, les commandes telles que **fteListAgents** échouent lorsqu'elles sont émises depuis le serveur de l'agent partenaire.

Topologie de base comportant un gestionnaire de files d'attente de coordination distinct et un agent partenaire

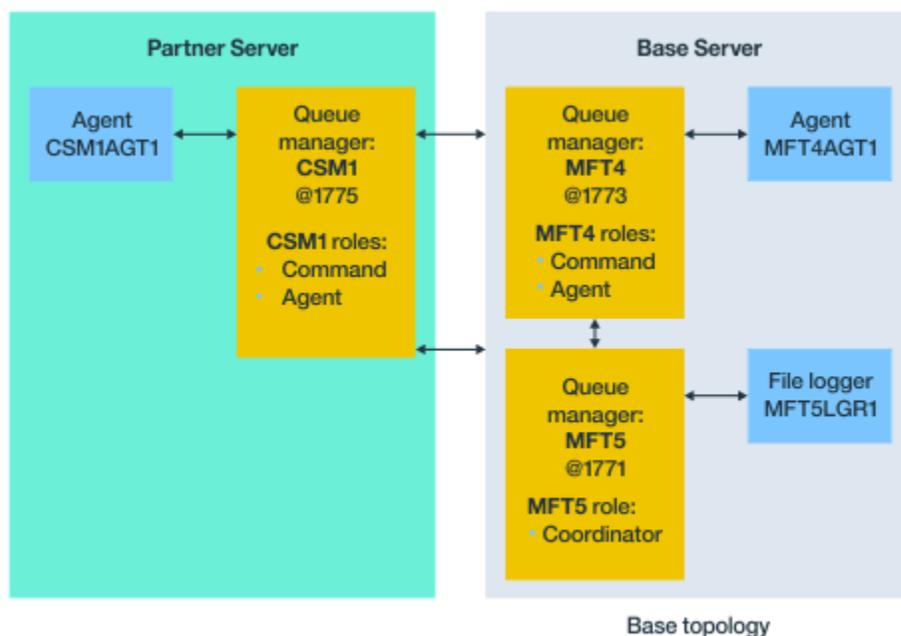


Figure 4. Topologie de base comportant un gestionnaire de files d'attente de coordination distinct et un agent partenaire

Dans la topologie de base présentée dans la figure 3, sur le serveur de base, le gestionnaire de files d'attente MFT4 est partagé pour les rôles de gestionnaire de file d'attente de commandes et de gestionnaire de file d'attente d'agent, et le gestionnaire de files d'attente MFT5 est dédié au rôle de gestionnaire de file d'attente de coordination.

La connectivité doit être établie entre tous les gestionnaires de files d'attente dans la topologie, y compris avec les gestionnaires de files d'attente de la topologie de base, MFT4 et MFT5.

Dans le gestionnaire de files d'attente sur le serveur partenaire, le gestionnaire de files d'attente CSM1 possède les rôles de gestionnaire de file d'attente de commandes et de gestionnaire de file d'attente d'agent.

Cette topologie peut échanger des fichiers entre les deux agents. Chaque agent partenaire doit se connecter à un gestionnaire de files d'attente, conformément au diagramme. Des agents partenaires supplémentaires peuvent être ajoutés de la même façon que le premier agent partenaire.

Topologie de base comportant un partenaire Managed File Transfer Agent

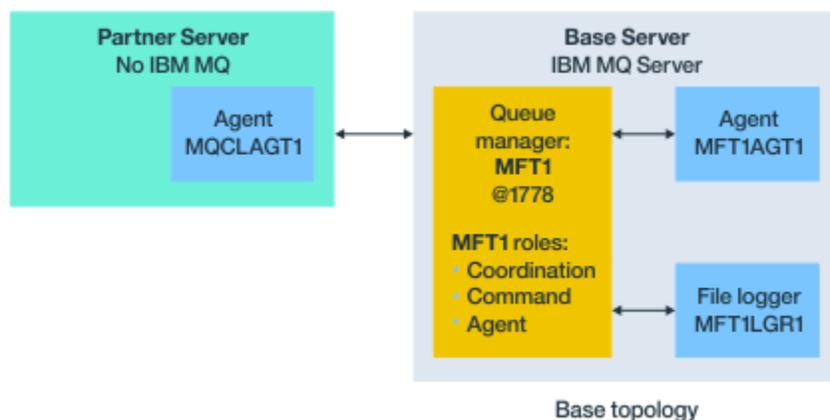


Figure 5. Topologie de base comportant un partenaire Managed File Transfer Agent

Cette topologie peut échanger des fichiers entre les deux agents.

Le serveur IBM MQ n'est pas installé sur le serveur dans l'agent partenaire, appelé MQCLAGT1 dans le diagramme.

L'agent partenaire est configuré à l'aide des mêmes commandes que pour le serveur IBM MQ installé, à quelques exceptions près :

- La configuration pour cet agent partenaire doit utiliser des connexions client IBM MQ au(x) gestionnaire(s) de files d'attente de base.
- Il n'est pas nécessaire d'exécuter les définitions IBM MQ de rôle de gestionnaire de file d'attente de coordination créées par les commandes de configuration sur le serveur de l'agent partenaire. Les définitions de gestionnaire de file d'attente de coordination existent déjà sur le serveur de base.

Toutefois, vous devez :

- Copier les définitions d'objet d'agent générées lors de la création de l'agent sur le serveur partenaire
- Transférer le fichier de définition sur le serveur de configuration de base
- Créer les définitions dans le gestionnaire de files d'attente identifié comme gestionnaire de file d'attente d'agent sur le serveur de base.

Dans ce cas, MFT1 assure les trois rôles et vous créez les objets pour l'agent MQCLAGT1 dans le gestionnaire de files d'attente MFT1.

Au lieu de copier les définitions d'objet sur le serveur de base, vous pouvez aussi exécuter la commande **fteDefine** pour l'agent MQCLAGT1 sur le serveur de base sur lequel se trouve le gestionnaire de file d'attente d'agent. Utilisez les définitions générées par la commande **fteDefine** pour créer les définitions d'agent requises dans le gestionnaire de file d'attente d'agent.

Par exemple, pour le diagramme présenté, vous devez copier MQCLAGT1_create.mqsc sur le serveur de configuration de base depuis le répertoire de l'agent qui se trouve sur le serveur partenaire, et créer les définitions d'agent requises dans le gestionnaire de files d'attente MFT1.

La configuration que vous effectuez sur le serveur de l'agent partenaire crée le répertoire de configuration Managed File Transfer et les fichiers de propriétés requis.

V 9.0.1 Sur le serveur partenaire, depuis IBM MQ 9.0.1, vous pouvez installer le client redistribuable Managed File Transfer depuis Fix Central.

Remarque : Le client redistribuable MQMFT diffère du client redistribuable IBM MQ en ce sens qu'il est déjà conditionné et qu'il ne nécessite pas l'utilisation de l'utilitaire **genmqpkg**. Pour plus d'informations, voir [Clients redistribuables](#).

Remarques relatives à la connectivité

Dans les diagrammes précédents, chaque ligne entre les agents et les gestionnaires de files d'attente représente une connexion à un gestionnaire de files d'attente.

Il peut s'agir :

- D'une connexion locale
- D'une connexion de canal de transmission de messages ou de liaisons
- D'une connexion MQI ou client IBM MQ

Le type de connexion que vous sélectionnez dans votre configuration dépend des paramètres que vous spécifiez.

- Lorsque vous spécifiez le paramètre de nom de gestionnaire de files d'attente avec d'autres paramètres de connexion, vous spécifiez une connexion de type liaisons.

Si le gestionnaire de files d'attente utilisé est local par rapport à la configuration Managed File Transfer, il représente également une connexion locale, s'il est utilisé sur le serveur de configuration de base.

- Si vous spécifiez le paramètre de nom de gestionnaire de files d'attente ainsi que les paramètres d'hôte, de port et de nom de canal correspondants, vous spécifiez une connexion client IBM MQ.

Lorsque des agents se trouvent sur le même hôte que le gestionnaire de file d'attente d'agent, une spécification de type liaisons, qui génère une connexion locale, est plus efficace.

Configuration du serveur de base

Cette rubrique explique comment configurer le serveur de base avec un gestionnaire de files d'attente de configuration distinct.

Avant de commencer

L'exemple suivant suppose que :

- Vous avez lu la section [«Remarques relatives à la connectivité»](#), à la page 33 et compris comment influencer le type de connexion aux gestionnaires de files d'attente dans la configuration.
- Vous disposez d'une infrastructure IBM MQ opérationnelle. Voir [Configuration des gestionnaires de files d'attente IBM MQ](#) pour plus d'informations.
- Les tâches de sécurité IBM MQ ont été effectuées.

Toutes les ressources système, telles que l'accès aux fichiers, sont configurées avec la sécurité appropriée.

Pour la configuration de la sécurité Managed File Transfer, voir [Présentation de la sécurité pour Managed File Transfer](#) et [Droits utilisateur sur les actions Managed File Transfer](#).

- Toutes les connexions IBM MQ sont testées une fois que IBM MQ est configuré à l'aide d'un exemple de programme pour envoyer et recevoir des messages, ou à l'aide d'un exemple de **amqscnxc** pour tester les connexions de type client IBM MQ.

L'exemple **amqscnxc** se connecte à un gestionnaire de files d'attente en définissant la connexion de canal dans l'exemple de code, qui est similaire à la manière dont Managed File Transfer se connecte lorsqu'il utilise une connexion de type MQI ou client IBM MQ.

- Ces instructions supposent qu'une version d'IBM MQ est installée sur le serveur que vous utilisez pour la configuration de base. S'il existe plusieurs installations d'IBM MQ sur le serveur de base, vous devez veiller à utiliser le chemin de fichier approprié pour la version d'IBM MQ que vous voulez utiliser.
- Les gestionnaires de files d'attente utilisés dans ces instructions ne requièrent pas d'authentification de connexion.

Bien qu'il soit plus simple d'effectuer une première configuration sans l'authentification de connexion, si votre entreprise requiert l'utilisation immédiate de l'authentification de connexion, voir [Managed File](#)

Transfer and IBM MQ connection authentication pour des instructions de configuration d'un fichier de données d'identification MQMFTCredentials.xml.

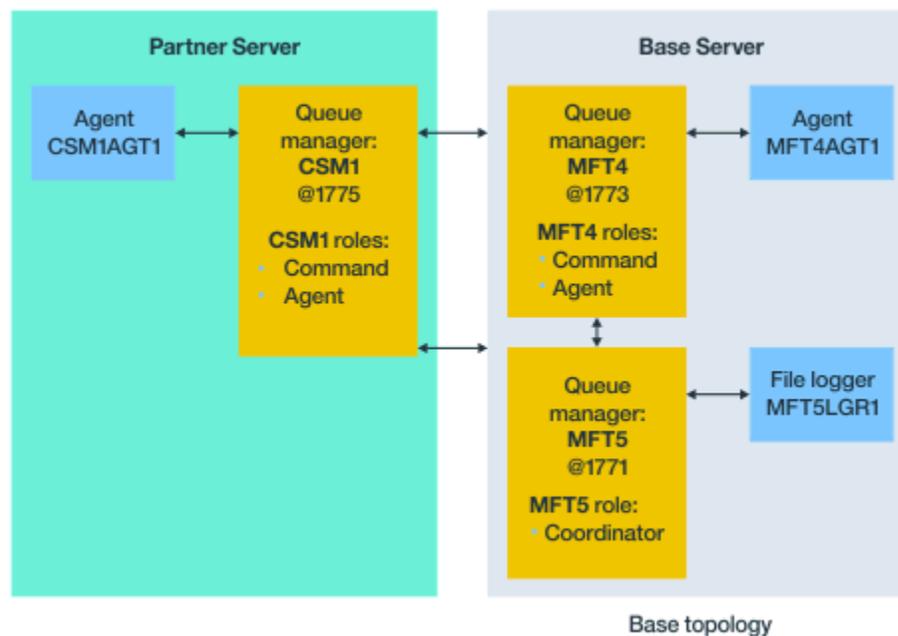


Figure 6. Topologie de base comportant un gestionnaire de files d'attente de coordination distinct et un agent partenaire

Pourquoi et quand exécuter cette tâche

Les rôles de gestionnaire de files d'attente pour l'exemple de configuration sont les suivants :

- Serveur de base
 - Le gestionnaire de files d'attente MFT5 est le gestionnaire de files d'attente de coordination.
 - Le gestionnaire de files d'attente MFT4 est utilisé comme gestionnaire de files d'attente d'agent pour l'agent MFT4AGT1, et assure également le rôle de gestionnaire de files d'attente de commandes pour la configuration MFT5 sur le serveur de base.
- Serveur partenaire
 - Le gestionnaire de files d'attente CSM1 double le gestionnaire de files d'attente d'agent pour l'agent CSM1AGT1 et le gestionnaire de files d'attente de commandes pour la configuration MFT5 sur le serveur partenaire.
 - Le gestionnaire de files d'attente MFT5, sur le serveur de base, est le gestionnaire de files d'attente de coordination.

Procédure

1. Configurez le gestionnaire de files d'attente de coordination.
2. Configurez le gestionnaire de files d'attente de commandes.
3. Configurez l'agent.
4. Configurez le consignateur.
5. Configurez un serveur partenaire.

Que faire ensuite

Configurez MQExplorer avec MQMFT pour pouvoir tester votre exemple de configuration.

Configuration du gestionnaire de file d'attente de coordination

Configuration du gestionnaire de file d'attente de coordination pour coordonner les transferts de fichiers.

Avant de commencer

Assurez-vous de disposer d'une connectivité complète entre les gestionnaires de files d'attente que vous avez configurés pour ce scénario.

Pourquoi et quand exécuter cette tâche

Cette tâche configure le gestionnaire de file d'attente de coordination MFT5, et les instructions présentées dans cette section supposent que vous utilisez une installation IBM MQ.

Si vous disposez de plusieurs installations, vous devez définir le chemin IBM MQ vers la version d'IBM MQ requise à l'aide de la commande `setmqenv` avant d'effectuer toute tâche de configuration.

Procédure

1. Connectez-vous en tant qu'administrateur Managed File Transfer.
2. Emettez la commande suivante pour identifier le gestionnaire de file d'attente de coordination et configurer la structure du répertoire de configuration :

```
fteSetupCoordination -coordinationQMgr MFT5
```

Répertoire du gestionnaire de file d'attente de coordination

```
C:\data\mqft\config\MFT5
```

Fichier **coordination.properties**

```
C:\data\mqft\config\MFT5\coordination.properties
```

La commande génère également un fichier de commandes MQSC que vous devez exécuter sur votre gestionnaire de files d'attente de coordination C:\data\mqft\config\MFT5\MFT5.mqsc:

3. Accédez au répertoire C:\data\mqft\config\MFT5.
4. Configurez le gestionnaire de files d'attente pour qu'il serve de gestionnaire de file d'attente de coordination en exécutant la commande ci-dessous.

Vous devez fournir le fichier de commandes MQSC généré par la commande que vous avez exécutée à l'étape «2», à la page 35 :

```
runmqsc MFT5 < MFT5.mqsc > mft5.txt
```

5. Ouvrez le fichier de résultats `mft5.txt` avec l'éditeur de votre choix. et vérifiez que les définitions ont été créées correctement.

Que faire ensuite

Configurez le [gestionnaire de file d'attente de commandes](#).

Configuration du gestionnaire de files d'attente de commandes

Configuration du gestionnaire de files d'attente de commandes.

Avant de commencer

Assurez-vous d'avoir configuré le gestionnaire de files d'attente de coordination. Pour plus d'informations, voir «[Configuration du gestionnaire de file d'attente de coordination](#)», à la page 35.

Pourquoi et quand exécuter cette tâche

Cette tâche identifie le gestionnaire de files d'attente de commandes.

Procédure

Entrez la commande suivante :

```
fteSetupCommands -connectionQMgr MFT4
```

Vous obtenez le message suivant BFGCL0245I: Le fichier C:\data\mqft\config\MFT4\command.properties a été créé avec succès.

Le gestionnaire de files d'attente de commandes ne requiert pas de définition IBM MQ supplémentaire. Après l'exécution de **fteSetupCommands**, le fichier command.properties est créé dans le répertoire de configuration MFT5.

Que faire ensuite

Configurez l'[agent](#).

Configuration de l'agent

Préparation d'un agent de transfert de fichier MFT4AGT1, y compris des scripts MQSC à exécuter.

Avant de commencer

Vous devez avoir configuré le gestionnaire de file d'attente de commandes. Pour plus d'informations, voir [«Configuration du gestionnaire de files d'attente de commandes»](#), à la page 35.

Pourquoi et quand exécuter cette tâche

Cette tâche prépare l'agent de transfert de fichier Windows MFT4AGT1.

Procédure

1. Entrez la commande suivante :

```
fteCreateAgent -agentName MFT4AGT1 -agentQMgr MFT4
```

Une fois que vous avez créé l'agent avec la commande **fteCreateAgent**, le répertoire agents et un sous-répertoire pour l'agent MFT4AGT1 sont ajoutés au répertoire MFT5.

Dans le répertoire `data\MFT5\agents\MFT4AGT1` se trouvent :

- Le fichier `agent.properties`
 - Le fichier `MFT4AGT1_create.mqsc`, qui contient les définitions IBM MQ requises par l'agent.
2. Accédez au répertoire `data\MFT5\agents\MFT4AGT1` et créez les définitions de gestionnaire de file d'attente d'agent requises en exécutant la commande suivante :

```
runmqsc MFT4 < MFT4AGT1_create.mqsc > mft4.txt
```

3. Ouvrez le fichier de résultats `mft4.txt` dans l'éditeur de votre choix et vérifiez que les définitions ont été créées correctement.
4. Démarrez l'agent en entrant la commande suivante : **fteStartAgent** MFT4AGT1.
5. Affichez l'agent en entrant la commande suivante : **fteListAgents**.

Une sortie similaire à la suivante apparaît :

```
5655-MFT, 5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
```

```
BFGPR0127W: No credentials file has been specified to connect to IBM MQ.  
Therefore, the assumption is that IBM MQ authentication has been disabled.  
Agent Name:      Queue Manager Name:    Status:  
MFT4AGT1        MFT4                      READY
```

Remarque : si vous n'avez pas activé l'authentification de connexion dans votre environnement Managed File Transfer, vous pouvez ignorer le message BFGPR0127W.

Si vous exécutez la commande **ftelistAgents** et recevez le message suivant, BFGCL0014W : Aucun agent ne correspond aux critères de sélection actuels., Pour plus d'informations, voir «[Que faire si votre agent MFT n'est pas répertorié par la commande fteListAgents ?](#)», à la page 309.

Que faire ensuite

Configurez le [consignateur](#).

Configuration du consigneur

Un consigneur de fichiers ou de base de données est requis pour conserver les informations d'historique et d'audit relatives à l'activité de transfert pour la configuration. Dans cet exemple, vous allez créer un consigneur de fichiers.

Avant de commencer

Vous devez avoir configuré :

- Le gestionnaire de files d'attente de configuration
- Gestionnaire de files d'attente de commandes
- L'agent

Procédure

1. Entrez la commande suivante :

```
fteCreateLogger -loggerQMgr MFT5 -loggerType FILE  
-fileLoggerMode CIRCULAR -fileSize 5MB -fileCount 3 MFT5lgr1
```

Après avoir exécuté la commande **fteCreateLogger**, le répertoire `data\mqft\config\MFT5\loggers` est créé avec un sous-répertoire MFT5LGR1.

Le sous-répertoire MFT5LGR1 contient le fichier `logger.properties`. Il contient également un fichier appelé `MFT5LGR1_create.mqsc`, qui comporte les définitions IBM MQ requises par le consigneur.

2. Accédez au répertoire `data\mqft\config\MFT5\loggers\MFT5LGR1`.
3. Exécutez le fichier de commandes MQSC associé :

```
runmqsc MFT5 < MFT5_create.mqsc
```

pour créer les définitions requises par le consigneur.

- a) Consultez les résultats des définitions d'objet pour vérifier que les objets requis ont été créés correctement.

4. Démarrez le consigneur avec la commande **fteStartLogger** MFT5LGR1.
5. Passez en revue le contenu du fichier `output0.log` dans `data\mqft\logs\MFT5\loggers\MFT5LGR1\logs`.

Normalement, après quelques informations sur le consigneur, la dernière instruction contient le message suivant : BFGDB0023I: Le consigneur a terminé les activités de démarrage et il est à présent en cours d'exécution.

Il peut arriver que les informations sur le consignateur ne soient pas écrites dans `output0.log`, lorsque le consignateur démarre pour la première fois. Si le fichier `output0.log` est vide, redémarrez le consignateur en entrant **fteStopLogger** MFT5LGR1 et en appuyant sur la touche **Entrée**.

Redémarrez le consignateur en entrant **fteStartLogger** MFTULGR1 et en appuyant sur la touche **Entrée**. A présent, le fichier `output0.log` contient des données.

Le même comportement peut se produire pour la version d'agent du fichier `output0.log`, lorsqu'un agent est démarré pour la première fois.

Démarrez et arrêtez l'agent avec les commandes **fteStopAgent** et **fteStartAgent**. Les données sont alors écrites dans le fichier `output0.log` de l'agent.

Résultats

Vous avez configuré le serveur de base, qui comprend le gestionnaire de file d'attente de coordination pour cette configuration.

Que faire ensuite

A présent, suivez la même procédure pour le serveur partenaire, qui comporte un agent distant.

Configuration d'un serveur partenaire

Configuration d'un serveur partenaire lorsque le serveur de base comporte un gestionnaire de files d'attente de coordination distinct

Avant de commencer

Assurez-vous d'avoir exécuté entièrement toutes les tâches de configuration d'un serveur de base, qui comporte un gestionnaire de files d'attente de configuration.

Pourquoi et quand exécuter cette tâche

Les mêmes postulats énoncés pour IBM MQ et la configuration de sécurité, ainsi que pour le chemin IBM MQ, s'appliquent au serveur partenaire.

Commencez par configurer le répertoire de configuration MFT5 et par identifier le gestionnaire de files d'attente de coordination avec la commande **fteSetupCoordination**.

Procédure

1. Créez le répertoire de configuration du serveur partenaire en émettant la commande suivante :

```
fteSetupCoordination -coordinationQMgr MFT5
-coordinationQMgrHost 177.16.20.15 -coordinationQMgrPort 1771
-coordinationQMgrChannel MQMFT.MFT5.SVRCONN
```

Remarques :

- a. Lorsque le gestionnaire de files d'attente de coordination se trouve sur un serveur différent de celui du serveur partenaire, la connexion au gestionnaire de files d'attente de coordination du serveur de base doit être définie comme une connexion client.

Si vous ne définissez pas la connexion au gestionnaire de files d'attente de coordination comme une connexion client IBM MQ, sur le serveur partenaire, toute commande Managed File Transfer qui établit la connexion au gestionnaire de files d'attente de coordination échoue.

fteListAgents est un exemple de commande établissant la connexion au gestionnaire de files d'attente de coordination.

b. Il n'est pas nécessaire de créer les définitions IBM MQ car les définitions requises par le gestionnaire de files d'attente de coordination sont créées lorsque vous configurez le serveur de base.

2. Identifiez le gestionnaire de files d'attente de commandes en émettant la commande suivante :

```
fteSetupCommands -connectionQMgr CSM1
```

Le gestionnaire de files d'attente de commandes ne requiert aucune définition IBM MQ supplémentaire.

3. Identifiez le gestionnaire de files d'attente d'agent partenaire et créez-le avec la commande suivante :

```
fteCreateAgent -agentName CSM1AGT1 -agentQMgr CSM1
```

4. Placez-vous dans le répertoire CSM1AGT1.

5. Créez les définitions IBM MQ requises par l'agent en émettant la commande suivante :

```
runmqsc CSM1 < CSM1AGT1_create.mqsc > csm1.txt
```

a) Ouvrez le fichier `csm1.txt` dans l'éditeur de votre choix pour vérifier que toutes les définitions requises par l'agent ont été créées correctement.

6. Démarrez l'agent avec la commande suivante :

```
fteStartAgent CSM1AGT1
```

7. Affichez l'agent en entrant **ftelistAgents**.

Une sortie similaire à la suivante apparaît :

```
C:\>ftelistAgents
5655-MFT, 5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
BFGPR0127W: No credentials file has been specified to connect to IBM MQ. Therefo
re, the assumption is that IBM MQ authentication has been disabled.
Agent Name:      Queue Manager Name:      Status:
CSM1AGT1        CSM1                          READY
MFT4AGT1        MFT4                          READY
```

Remarque : si vous n'avez pas activé l'authentification de connexion dans votre environnement Managed File Transfer, vous pouvez ignorer le message BFGPR0127W.

Si vous exécutez la commande **ftelistAgents** et recevez le message suivant, BFGCL0014W : Aucun agent ne correspond aux critères de sélection actuels., Pour plus d'informations, voir «Que faire si votre agent MFT n'est pas répertorié par la commande `ftelistAgents` ?», à la page 309.

Si le statut de l'un des agents est UNREACHABLE, voir «Que faire si un agent est affiché comme étant à l'état UNKNOWN (inconnu)», à la page 311 pour plus d'informations.

Configuration d'IBM MQ Explorer avec MFT

Cette tâche permet de connecter IBM MQ Explorer à la configuration Managed File Transfer.

Procédure

1. Démarrez IBM MQ Explorer.

2. Dans le panneau Navigator de gauche, faites défiler vers le bas et développez le dossier: Managed File Transfer.

L'entrée du gestionnaire de file d'attente de coordination, MFT5, apparaît.

3. Cliquez avec le bouton droit de la souris sur MFT5 et sélectionnez **Connexion**.

a) Sélectionnez Agents dans le menu déroulant qui apparaît et assurez-vous que les deux agents, MFT4AGT1 et CSMAGT1, sont à l'état Ready (Prêt).

Que faire ensuite

Testez votre exemple de configuration avec [IBM MQ Explorer](#).

Utilisation d'IBM MQ Explorer pour tester le transfert de fichier

Cette tâche fournit un exemple d'utilisation de IBM MQ Explorer avec Managed File Transfer, pour tester un transfert de fichier, après avoir configuré IBM MQ Explorer comme indiqué dans la rubrique précédente.

Avant de commencer

Vérifiez que vous disposez d'un système opérationnel, que les agents sont READY et que IBM MQ Explorer fonctionne. Pour plus d'informations, voir «[Configuration d'IBM MQ Explorer avec MFT](#)», à la page 39.

Pourquoi et quand exécuter cette tâche

Identifiez le fichier à utiliser pour tester le transfert et déterminez dans quel répertoire le copier. Dans cet exemple, le fichier `test-file.txt` qui se trouve dans le répertoire `C:\temp\mft` est utilisé.

```
C:\temp\mft> dir *
Date stamp 61 test-file.txt
1 File(s) 61 bytes
```

Procédure

1. Démarrez IBM MQ Explorer sous Windows.
2. Dans le panneau Navigator de gauche, développez le dossier: **Managed File Transfer**.
L'entrée du gestionnaire de file d'attente de coordination, MFT5, apparaît.
3. Cliquez avec le bouton droit de la souris sur MFT5 et sélectionnez **Connexion**.
4. Une fois la connexion établie, cliquez avec le bouton droit de la souris sur MFT5 et sélectionnez **Nouveau transfert**.
 - a) Utilisez le menu déroulant afin de sélectionner MFT4AGT1 pour l'agent source et CSMAGT1 pour l'agent de destination.
 - b) Cliquez sur **Suivant**.
 - c) Cliquez sur **Ajouter** dans la fenêtre suivante.
Une grande boîte de dialogue s'ouvre. Le côté gauche est dédié à la **Source** et le côté droit à la **Destination**.
5. Dans le panneau **Source** :
 - a) Sélectionnez **Transfert de texte** car le fichier est un fichier texte.
 - b) Sélectionnez **Parcourir** pour localiser le fichier.
Dans ce cas, le fichier s'appelle `C:\temp\mft\test-file.txt`.



Avertissement : ne cliquez pas sur **OK** car vous devez remplir le panneau **Destination**.

6. Dans le panneau **Destination** :
 - a) Entrez le nom que vous voulez attribuer au fichier dans la destination, par exemple `test-file.txt`.
L'utilisation de chemins d'accès relatifs est prise en charge. La première partie du chemin d'accès complet correspond au répertoire de base de l'ID utilisateur qui démarre l'agent de destination.
 - b) Sélectionnez `Overwrite files if present` si vous avez besoin de cette option.
 - c) Cliquez sur **OK**.
Le fichier que vous avez sélectionné apparaît dans le panneau des **nouveaux transferts**.

7. Si le menu de configuration de MFT5 est fermé et se présente sous la forme +MFT5, développez-le en cliquant sur le signe +.
8. Restez dans la configuration MFT sélectionnée.
Ensuite, vérifiez le statut du transfert comme suit.
9. Cliquez sur **Journal de transfert** sous le gestionnaire de file d'attente de coordination MFT5.
10. Examinez le statut dans Managed File Transfer - Current Transfer progress panel, immédiatement en dessous du panneau supérieur **Journal de transfert** et attendez que le transfert soit terminé.
Si le transfert a abouti (l'arrière-plan est vert), cela signifie que le test de votre configuration est réussi.
Si le transfert a échoué (l'arrière-plan est rouge), cela signifie qu'une erreur est survenue.
Dans la plupart des cas, vous pouvez utiliser la barre de défilement située sous le panneau supérieur **Journal de transfert** et afficher un récapitulatif des motifs de l'échec.
 - a) Si vous ne parvenez pas à déterminer la raison pour laquelle le transfert a échoué, cliquez deux fois sur l'entrée correspondant au transfert dans le panneau supérieur **Journal de transfert**.
 - b) Sélectionnez XML dans le panneau de gauche de la fenêtre en incrustation qui apparaît.
 - c) Faites défiler les informations afin de déterminer la cause de l'erreur.
 - d) Apportez les corrections nécessaires et testez à nouveau le transfert.

Sécurisation de MFT

Juste après l'installation et si vous n'avez apporté aucune modification, Managed File Transfer présente un niveau de sécurité pouvant être adapté à des fins de test ou d'évaluation dans un environnement protégé. Toutefois, dans un environnement de production, vous devez envisager de contrôler de façon appropriée les utilisateurs pouvant démarrer des opérations de transfert de fichier et lire et écrire les fichiers transférés, et déterminer comment protéger l'intégrité des fichiers.

Concepts associés

«MFT bacs à sable», à la page 46

Vous pouvez restreindre la zone du système de fichiers à laquelle l'agent peut accéder dans le cadre d'un transfert. La zone à laquelle l'agent est limité est appelée le bac à sable. Vous pouvez appliquer des restrictions à l'agent ou à l'utilisateur qui demande un transfert.

Tâches associées

«Configuration du chiffrement SSL ou TLS pour MFT», à la page 52

Vous pouvez utiliser SSL ou TLS avec IBM MQ Managed File Transfer pour sécuriser les communications entre les agents et leurs gestionnaires de files d'attente d'agent, les commandes et les gestionnaires de files d'attente auxquels ils se connectent, ainsi que les différentes connexions entre les gestionnaires de files d'attente et les gestionnaires de files d'attente dans votre topologie.

Référence associée

«Restriction des droits de groupe pour les ressources spécifiques à MFT», à la page 362

Au lieu d'accorder des droits à des utilisateurs individuels pour tous les différents objets pouvant être impliqués, configurez deux groupes de sécurité pour l'administration du contrôle d'accès Managed File Transfer : FTEUSER et FTEAGENT. Il incombe à l'administrateur IBM MQ de créer et de remplir ces groupes. L'administrateur peut choisir d'étendre ou de modifier la configuration proposée décrite ici.

«Droits de gestion pour les ressources spécifiques à MFT», à la page 361

Pour toute demande de transfert de fichier, les processus de l'agent Managed File Transfer requièrent un certain niveau d'accès à leurs systèmes de fichiers locaux. En outre, l'ID utilisateur associé au processus d'agent et les ID utilisateur associés aux utilisateurs effectuant des opérations de transfert de fichiers doivent être autorisés à utiliser certains objets IBM MQ .

«Droits d'accès de MFT aux systèmes de fichiers», à la page 374

Pour toute demande de transfert de fichier, les processus de l'agent Managed File Transfer requièrent un certain niveau d'accès à leurs systèmes de fichiers locaux.

«Authentification de connexion MFT et IBM MQ», à la page 42

L'authentification de connexion permet à un gestionnaire de files d'attente d'être configuré pour authentifier les applications à l'aide d'un ID utilisateur et d'un mot de passe fournis. Si la sécurité est activée pour le gestionnaire de files d'attente associé et que les données d'identification (ID utilisateur et mot de passe) sont requises, la fonction d'authentification de connexion doit être activée pour que la connexion à un gestionnaire de files d'attente puisse être établie. L'authentification de connexion peut être exécutée en mode compatibilité ou en mode d'authentification MQCSP.

«Propriété commandPath MFT», à la page 374

La propriété commandPath permet de spécifier les emplacements à partir desquels Managed File Transfer peut exécuter des commandes. Soyez extrêmement prudent lorsque vous définissez cette propriété car toute commande de l'un des emplacements commandPaths spécifiés peut être effectivement appelée à partir d'un système client distant qui peut envoyer des commandes à l'agent.

«Droits de publication du journal et des messages d'état des agents MFT», à la page 373

Les agents Managed File Transfer émettent divers messages de journal, de progression et de statut qui sont publiés sur le gestionnaire de files d'attente de coordination. La publication de ces messages est soumise au modèle de sécurité IBM MQ et, dans certains cas, vous devrez peut-être effectuer une configuration supplémentaire pour activer la publication.

Authentification de connexion MFT et IBM MQ

L'authentification de connexion permet à un gestionnaire de files d'attente d'être configuré pour authentifier les applications à l'aide d'un ID utilisateur et d'un mot de passe fournis. Si la sécurité est activée pour le gestionnaire de files d'attente associé et que les données d'identification (ID utilisateur et mot de passe) sont requises, la fonction d'authentification de connexion doit être activée pour que la connexion à un gestionnaire de files d'attente puisse être établie. L'authentification de connexion peut être exécutée en mode compatibilité ou en mode d'authentification MQCSP.

Méthodes de fourniture des détails des données d'identification

De nombreuses commandes Managed File Transfer prennent en charge les méthodes suivantes pour fournir les détails des données d'identification:

Détails fournis par les arguments de ligne de commande.

Les détails des données d'identification peuvent être spécifiés à l'aide des paramètres **-mquserid** et **-mqpassword**. Si le **-mqpassword** n'est pas fourni, l'utilisateur est invité à indiquer le mot de passe dans lequel l'entrée n'est pas affichée.

Détails fournis depuis un fichier de données d'identification : **MQMFTCredentials.xml**.

Les données d'identification détaillées peuvent être prédéfinies dans un fichier **MQMFTCredentials.xml** sous forme de texte en clair ou brouillé.

Pour plus d'informations sur la configuration d'un fichier **MQMFTCredentials.xml** sur IBM MQ for Multiplatforms, voir «Configuration de **MQMFTCredentials.xml** sur plusieurs plateformes», à la page 43.

Pour plus d'informations sur la configuration d'un fichier **MQMFTCredentials.xml** sur IBM MQ for z/OS, voir [Configuration de **MQMFTCredentials.xml** sur z/OS](#).

Priorité

L'ordre de priorité des méthodes pour déterminer les données d'identification détaillées est le suivant :

1. Argument de ligne de commande
2. Index **MQMFTCredentials.xml** par gestionnaire de files d'attente associé et utilisateur exécutant la commande
3. Index **MQMFTCredentials.xml** par gestionnaire de files d'attente associé

4. Mode de compatibilité amont par défaut dans lequel aucun détail de données d'identification n'est fourni pour permettre la compatibilité avec les versions précédentes de IBM MQ ou IBM WebSphere MQ

Remarques :

- Les commandes **fteStartAgent** et **fteStartLogger** ne prennent pas en charge l'argument de ligne de commande **-mquserid** ou **-mqpassword**, et les données d'identification détaillées ne peuvent être spécifiées qu'à l'aide du fichier `MQMFTCredentials.xml`.

z/OS

Sous z/OS, le mot de passe doit être en majuscules, même si le mot de passe de l'utilisateur est en minuscules. Par exemple, si le mot de passe de l'utilisateur est "motdepasse", il doit être entré sous la forme "MOTDEPASSE".

Référence associée

Quelle commande MFT se connecte à quel gestionnaire de files d'attente

Une topologie Managed File Transfer se compose d'un certain nombre de composants différents.

Format du fichier de données d'identification MFT

Le fichier `MQMFTCredentials.xml` contient les informations d'ID utilisateur et de mot de passe sensibles. Les éléments du fichier `MQMFTCredentials.xml` doivent être conformes au schéma `MQMFTCredentials.xsd`. La sécurité des fichiers de données d'identification est de la responsabilité de l'utilisateur.

Configuration de `MQMFTCredentials.xml` sur plusieurs plateformes

Si Managed File Transfer (MFT) est configuré avec la sécurité activée, l'authentification de connexion 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. De même, les consigneurs MFT peuvent être tenus de spécifier un ID utilisateur et un mot de passe lors de la connexion à une base de données. Ces données d'identification peuvent être stockées dans le fichier de données d'identification MFT.

Pourquoi et quand exécuter cette tâche

Les éléments du fichier `MQMFTCredentials.xml` doivent être conformes au schéma `MQMFTCredentials.xsd`. Pour plus d'informations sur le format de `MQMFTCredentials.xml`, voir [Format de fichier des données d'identification MFT](#).

Vous trouverez un exemple de fichier de données d'identification dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/credentials`.

Vous pouvez disposer d'un fichier de données d'identification MFT pour le gestionnaire de files d'attente de coordination, d'un fichier pour le gestionnaire de files d'attente de commandes, d'un fichier pour chaque agent et d'un fichier pour chaque consigneur. Vous pouvez également disposer d'un fichier qui est utilisé par tous les éléments de votre topologie.

L'emplacement par défaut du fichier de données d'identification MFT est le suivant:

Linux → UNIX **UNIX and Linux**
\$HOME

Windows **Windows**
%USERPROFILE% ou %HOMEDRIVE%%HOMEPATH%

Si le fichier de données d'identification est stocké à un autre emplacement, vous pouvez utiliser les propriétés suivantes pour indiquer où les commandes doivent le rechercher:

Tableau 1. : Propriétés qui définissent l'emplacement du fichier MQMFTCredentials.xml pour diverses commandes.

Type de commande	Fichier de propriétés	Nom de la propriété
Commande qui se connecte au gestionnaire de file d'attente de coordination	coordination.properties	coordinationQMGrAuthenticationCredentialsFile
Commande qui se connecte au gestionnaire de files d'attente de commandes	connection.properties	connectionQMGrAuthenticationCredentialsFile
Commande qui se connecte à un processus d'agent	agent.properties	agentQMGrAuthenticationCredentialsFile
Commande qui se connecte à un processus de consignateur	logger.properties	loggerQMGrAuthenticationCredentialsFile

Tableau 2. : Propriétés qui définissent l'emplacement du fichier MQMFTCredentials.xml pour les agents et les processus de consignateur.

Type de commande	Fichier de propriétés	Nom de la propriété
Agents MFT	agent.properties	agentQMGrAuthenticationCredentialsFile
MFT loggers	logger.properties	loggerQMGrAuthenticationCredentialsFile

Pour plus de détails sur les commandes et les processus qui se connectent à quel gestionnaire de files d'attente, voir [Quelles commandes et quels processus MFT se connectent à quel gestionnaire de files d'attente.](#)

Etant donné que le fichier de données d'identification contient des informations d'ID utilisateur et de mot de passe, il requiert des droits spéciaux pour empêcher tout accès non autorisé à ce fichier:

Linux → UNIX **UNIX and Linux**

```
chown <agent owner userid>
chmod 600
```

Windows **Windows**

Vérifiez que l'héritage n'est pas activé, puis supprimez tous les ID utilisateur à l'exception de ceux qui exécutent l'agent ou le consignateur qui utiliseront le fichier de données d'identification.

Les données d'identification utilisées pour la connexion à un gestionnaire de files d'attente de coordination MFT , dans le plug-in IBM MQ Explorer Managed File Transfer , dépendent du type de configuration:

Global (configuration sur disque local)

Une configuration globale utilise le fichier de données d'identification spécifié dans les propriétés de coordination et de commande.

Local (défini dans IBM MQ Explorer):

Une configuration locale utilise les propriétés des détails de connexion du gestionnaire de files d'attente associé dans IBM MQ Explorer.

Tâches associées

«Activation de l'authentification MQCSP», à la page 45

Depuis IBM MQ 9.0.3 for Continuous Delivery et IBM MQ 9.0.0 Fix Pack 2 for Long Term Support, vous pouvez activer le mode d'authentification MQCSP pour l'authentification de connexion du plug-in IBM MQ Explorer MFT se connectant à un gestionnaire de files d'attente de coordination ou à un gestionnaire de files d'attente de commandes. Vous pouvez également activer le mode d'authentification MQCSP pour l'authentification de connexion pour un agent Managed File Transfer se connectant à un gestionnaire de files d'attente de coordination ou à un gestionnaire de files d'attente de commandes.

Référence associée

Format du fichier de données d'identification MFT

Le fichier `MQMFTCcredentials.xml` contient les informations d'ID utilisateur et de mot de passe sensibles. Les éléments du fichier `MQMFTCcredentials.xml` doivent être conformes au schéma `MQMFTCcredentials.xsd`. La sécurité des fichiers de données d'identification est de la responsabilité de l'utilisateur.

Information associée

fteObfuscate: chiffrement des données sensibles

Activation de l'authentification MQCSP

Depuis IBM MQ 9.0.3 for Continuous Delivery et IBM MQ 9.0.0 Fix Pack 2 for Long Term Support, vous pouvez activer le mode d'authentification MQCSP pour l'authentification de connexion du plug-in IBM MQ Explorer MFT se connectant à un gestionnaire de files d'attente de coordination ou à un gestionnaire de files d'attente de commandes. Vous pouvez également activer le mode d'authentification MQCSP pour l'authentification de connexion pour un agent Managed File Transfer se connectant à un gestionnaire de files d'attente de coordination ou à un gestionnaire de files d'attente de commandes.

Pourquoi et quand exécuter cette tâche

Si vous utilisez le plug-in IBM MQ Explorer Managed File Transfer ou que des agents Managed File Transfer se connectent à un gestionnaire de files d'attente à l'aide du transport CLIENT et que vous spécifiez un mot de passe, l'agent ne s'authentifie pas auprès du gestionnaire de files d'attente si le mot de passe spécifié comporte plus de 12 caractères. En effet, le code n'utilise pas l'authentification MQCSP et s'authentifie en mode compatibilité, ce qui limite la longueur du mot de passe à 12 caractères.

Depuis IBM MQ 9.0.3 et IBM MQ 9.0.0 Fix Pack 2, vous pouvez désactiver le mode de compatibilité par défaut et activer le mode d'authentification MQCSP.

Procédure

- Pour désactiver le mode compatibilité et activer l'authentification MQCSP pour un gestionnaire de files d'attente de coordination ou un gestionnaire de files d'attente de commandes dans IBM MQ Explorer, procédez comme suit:
 - a) Sélectionnez le gestionnaire de files d'attente auquel vous souhaitez vous connecter.
 - b) Cliquez avec le bouton droit de la souris et sélectionnez **Détails de connexion-> Propriétés** dans le menu contextuel.
 - c) Cliquez sur l'onglet **ID utilisateur**.
 - d) Vérifiez que l'option **Activer l'identification de l'utilisateur** est sélectionnée et désélectionnez la case à cocher **Mode de compatibilité d'identification de l'utilisateur**.
- Pour désactiver le mode compatibilité et activer l'authentification MQCSP pour un agent Managed File Transfer, ajoutez le paramètre **useMQCSPAuthentication** au fichier `MQMFTCcredentials.xml` pour l'utilisateur concerné et définissez-le sur `true`.

Le paramètre doit être défini sur `true`. Si le paramètre n'est pas spécifié, il est défini par défaut sur `false` et le mode compatibilité est utilisé pour authentifier l'utilisateur auprès du gestionnaire de files d'attente.

L'exemple suivant montre comment définir le paramètre **useMQCSPAAuthentication** dans le fichier `MQMFTCredentials.xml` :

```
<tns:qmgr name="CoordQueueMgr" user="ernest" mqUserId="ernest"
mqPassword="AveryL0ngPassw0rd2135" useMQCSPAAuthentication="true"/>
```

Référence associée

«Authentification de connexion MFT et IBM MQ», à la page 42

L'authentification de connexion permet à un gestionnaire de files d'attente d'être configuré pour authentifier les applications à l'aide d'un ID utilisateur et d'un mot de passe fournis. Si la sécurité est activée pour le gestionnaire de files d'attente associé et que les données d'identification (ID utilisateur et mot de passe) sont requises, la fonction d'authentification de connexion doit être activée pour que la connexion à un gestionnaire de files d'attente puisse être établie. L'authentification de connexion peut être exécutée en mode compatibilité ou en mode d'authentification MQCSP.

«Format de fichier de données d'identification MFT», à la page 835

Le fichier `MQMFTCredentials.xml` contient les informations d'ID utilisateur et de mot de passe sensibles. Les éléments du fichier `MQMFTCredentials.xml` doivent être conformes au schéma `MQMFTCredentials.xsd`. La sécurité des fichiers de données d'identification est de la responsabilité de l'utilisateur.

Information associée

Protection par mot de passe MQCSP

MFT bacs à sable

Vous pouvez restreindre la zone du système de fichiers à laquelle l'agent peut accéder dans le cadre d'un transfert. La zone à laquelle l'agent est limité est appelée le bac à sable. Vous pouvez appliquer des restrictions à l'agent ou à l'utilisateur qui demande un transfert.

Les bacs à sable ne sont pas pris en charge lorsque l'agent est un agent de pont de protocole ou un agent de pont Connect:Direct. Vous ne pouvez pas utiliser le bac à sable d'agent pour les agents qui doivent effectuer un transfert vers ou depuis des files d'attente IBM MQ.

Référence associée

«Utilisation des bacs à sable d'agent MFT», à la page 46

Pour ajouter un niveau de sécurité supplémentaire à Managed File Transfer, vous pouvez restreindre la zone d'un système de fichiers à laquelle un agent peut accéder.

«Utilisation des bacs à sable utilisateur MFT», à la page 48

Vous pouvez restreindre la zone du système de fichiers dans laquelle les fichiers peuvent être transférés en fonction du nom d'utilisateur MQMD qui demande le transfert.

Utilisation des bacs à sable d'agent MFT

Pour ajouter un niveau de sécurité supplémentaire à Managed File Transfer, vous pouvez restreindre la zone d'un système de fichiers à laquelle un agent peut accéder.

Vous ne pouvez pas utiliser le bac à sable d'agent pour les agents qui sont transférés vers ou depuis des files d'attente IBM MQ. La restriction de l'accès aux files d'attente IBM MQ avec bac à sable peut être implémentée à la place en utilisant le bac à sable utilisateur, qui est la solution recommandée pour toutes les exigences en matière de bac à sable. Pour plus d'informations sur le bac à sable utilisateur, voir «Utilisation des bacs à sable utilisateur MFT», à la page 48

Pour activer le bac à sable de l'agent, ajoutez la propriété suivante au fichier `agent.properties` de l'agent que vous souhaitez restreindre:

```
sandboxRoot=[!]restricted_directory_nameseparator...separator[!]restricted_directory_name
```

où :

- *restricted_directory_name* est un chemin de répertoire à autoriser ou à refuser.
- **!** est facultatif et indique que la valeur suivante pour *restricted_directory_name* est refusée (exclue). Si **!** n'est pas spécifié, *restricted_directory_name* est un chemin d'accès autorisé (inclus).
- *separator* est le séparateur spécifique à la plateforme.

Par exemple, si vous souhaitez restreindre l'accès de AGENT1 au répertoire /tmp uniquement, mais ne pas autoriser l'accès au sous-répertoire `private`, définissez la propriété comme suit dans le fichier `agent.properties` appartenant à AGENT1: `sandboxRoot=/tmp:!/tmp/private`.

La propriété `sandboxRoot` est décrite dans [Propriétés avancées de l'agent](#).

Les bacs à sable d'agent et d'utilisateur ne sont pas pris en charge sur les agents de pont de protocole ou sur les agents de pont Connect:Direct.

Utilisation d'un bac à sable sur les plateformes UNIX, Linux et Windows

ULW Sur les plateformes UNIX, Linux et Windows, le bac à sable restreint les répertoires dans lesquels un Managed File Transfer Agent peut effectuer des opérations de lecture et d'écriture. Lorsque le bac à sable est activé, le Managed File Transfer Agent peut lire et écrire dans les répertoires spécifiés comme étant autorisés, ainsi que dans les sous-répertoires que les répertoires spécifiés contiennent, sauf si les sous-répertoires sont spécifiés comme étant refusés dans `sandboxRoot`. Le bac à sable Managed File Transfer n'est pas prioritaire sur la sécurité du système d'exploitation. L'utilisateur qui a démarré Managed File Transfer Agent doit disposer de l'accès de niveau système d'exploitation approprié à n'importe quel répertoire pour pouvoir lire ou écrire dans le répertoire. Un lien symbolique vers un répertoire n'est pas suivi si le répertoire auquel il est lié se trouve en dehors des répertoires `sandboxRoot` spécifiés (et des sous-répertoires).

Utilisation d'un bac à sable sous z/OS

z/OS Sous z/OS, le bac à sable restreint les qualificatifs de nom de fichier que le Managed File Transfer Agent peut lire et dans lesquels il peut écrire. L'utilisateur qui a démarré Managed File Transfer Agent doit disposer des droits d'accès appropriés au système d'exploitation pour tous les fichiers concernés. Si vous placez une valeur de qualificatif de nom de fichier `sandboxRoot` entre guillemets, la valeur suit la convention z/OS normale et est traitée comme qualifiée complète. Si vous omettez les guillemets, `sandboxRoot` est préfixé avec l'ID utilisateur en cours. Par exemple, si vous définissez la propriété `sandboxRoot` sur la valeur suivante: `sandboxRoot=//test`, l'agent peut accéder aux ensembles de données suivants (en notation z/OS standard) `//username.test.**`. Au moment de l'exécution, si les niveaux initiaux du nom de fichier entièrement résolu ne correspondent pas à `sandboxRoot`, la demande de transfert est rejetée.

Utilisation d'un bac à sable sur des systèmes IBM i

IBM i Pour les fichiers du système de fichiers intégré sur les systèmes IBM i, le bac à sable restreint les répertoires dans lesquels un Managed File Transfer Agent peut effectuer des opérations de lecture et d'écriture. Lorsque le bac à sable est activé, le Managed File Transfer Agent peut lire et écrire dans les répertoires spécifiés comme étant autorisés, ainsi que dans les sous-répertoires que les répertoires spécifiés contiennent, sauf si les sous-répertoires sont spécifiés comme étant refusés dans `sandboxRoot`. Le bac à sable Managed File Transfer n'est pas prioritaire sur la sécurité du système d'exploitation. L'utilisateur qui a démarré Managed File Transfer Agent doit disposer de l'accès de niveau système d'exploitation approprié à n'importe quel répertoire pour pouvoir lire ou écrire dans le répertoire. Un lien symbolique vers un répertoire n'est pas suivi si le répertoire auquel il est lié se trouve en dehors des répertoires `sandboxRoot` spécifiés (et des sous-répertoires).

Référence associée

«Vérifications supplémentaires pour les transferts de caractères génériques», à la page 51

Depuis IBM MQ 9.0.1 dans Continuous Delivery et IBM MQ 9.0.0 Fix Pack 1 dans Long Term Support, si un agent a été configuré avec un bac à sable d'utilisateur ou d'agent afin de restreindre les emplacements vers et depuis lesquels l'agent peut transférer des fichiers, vous pouvez spécifier que des vérifications supplémentaires doivent être effectuées sur les transferts de caractères génériques pour cet agent.

«Utilisation des bacs à sable d'agent MFT», à la page 46

Pour ajouter un niveau de sécurité supplémentaire à Managed File Transfer, vous pouvez restreindre la zone d'un système de fichiers à laquelle un agent peut accéder.

«Le fichier MFT agent.properties», à la page 564

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.

Utilisation des bacs à sable utilisateur MFT

Vous pouvez restreindre la zone du système de fichiers dans laquelle les fichiers peuvent être transférés en fonction du nom d'utilisateur MQMD qui demande le transfert.

Les bacs à sable utilisateur ne sont pas pris en charge lorsque l'agent est un agent de pont de protocole ou un agent de pont Connect:Direct .

Pour activer le bac à sable utilisateur, ajoutez la propriété suivante au fichier `agent.properties` de l'agent que vous souhaitez restreindre:

```
userSandboxes=true
```

Lorsque cette propriété est présente et définie sur `true`, l'agent utilise les informations du fichier `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name/UserSandboxes.xml` pour déterminer les parties du système de fichiers auxquelles l'utilisateur qui demande le transfert peut accéder.

Le XML `UserSandboxes.xml` est composé d'un élément `<agent>` qui contient zéro ou plusieurs éléments `<sandbox>` . Ces éléments décrivent quelles règles sont appliquées à quels utilisateurs. L'attribut `user` de l'élément `<sandbox>` est un modèle utilisé pour établir une correspondance avec l'utilisateur MQMD de la demande.

Le fichier `UserSandboxes.xml` est rechargé périodiquement par l'agent et toute modification valide apportée au fichier aura une incidence sur le comportement de l'agent. L'intervalle de rechargement par défaut est de 30 secondes. Vous pouvez modifier cet intervalle en spécifiant la propriété d'agent `xmlConfigReloadInterval` dans le fichier `agent.properties` .

Si vous spécifiez l'attribut ou la valeur `userPattern="regex"` , l'attribut `user` est interprété comme une expression régulière Java . Pour plus d'informations, voir la section «Expressions régulières utilisées par MFT», à la page 679.

Si vous ne spécifiez pas l'attribut ou la valeur `userPattern="regex"` , l'attribut `user` est interprété comme un modèle avec les caractères génériques suivants:

- astérisque (*), qui représente zéro ou plusieurs caractères
- point d'interrogation (?), qui représente exactement un caractère

Les correspondances sont effectuées dans l'ordre dans lequel les éléments `<sandbox>` sont répertoriés dans le fichier. Seule la première correspondance est utilisée, toutes les correspondances potentielles suivantes dans le fichier sont ignorées. Si aucun des éléments `<sandbox>` spécifiés dans le fichier ne correspond à l'utilisateur MQMD associé au message de demande de transfert, le transfert ne peut pas accéder au système de fichiers. Lorsqu'une correspondance a été trouvée entre le nom d'utilisateur MQMD et un attribut `user` , la correspondance identifie un ensemble de règles dans un élément `<sandbox>` qui sont appliquées au transfert. Cet ensemble de règles est utilisé pour déterminer quels fichiers, ou ensembles de données, peuvent être lus ou écrits dans le cadre du transfert.

Chaque ensemble de règles peut spécifier un élément `<read>` , qui identifie les fichiers qui peuvent être lus, et un élément `<write>` , qui identifie les fichiers qui peuvent être écrits. Si vous omettez les éléments `<read>` ou `<write>` d'un ensemble de règles, il est supposé que l'utilisateur associé à cet ensemble de règles n'est pas autorisé à effectuer des lectures ou des écritures, selon le cas.

Remarque : L'élément `<read>` doit être antérieur à l'élément `<write>` et l'élément `<include>` doit être antérieur à l'élément `<exclude>`, dans le fichier `UserSandboxes.xml`.

Chaque élément `<read>` ou `<write>` contient un ou plusieurs canevas utilisés pour déterminer si un fichier se trouve dans le bac à sable et peut être transféré. Spécifiez ces modèles à l'aide des éléments `<include>` et `<exclude>`. L'attribut `name` de l'élément `<include>` ou `<exclude>` spécifie le modèle à mettre en correspondance. Un attribut `type` facultatif indique si la valeur de nom est un fichier ou un modèle de file d'attente. Si l'attribut `type` n'est pas spécifié, l'agent traite le modèle comme un modèle de chemin de fichier ou de répertoire. Exemple :

```
<tns:read>
  <tns:include name="/home/user/**"/>
  <tns:include name="USER.**" type="queue"/>
  <tns:exclude name="/home/user/private/**"/>
</tns:read>
```

Les modèles `<include>` et `<exclude>` `name` sont utilisés par l'agent pour déterminer si les fichiers, les ensembles de données, ou les files d'attente peuvent être lus ou écrits. Une opération est autorisée si le chemin de fichier canonique, l'ensemble de données ou le nom de file d'attente correspond à au moins un des modèles inclus et à exactement zéro des modèles exclus. Les modèles spécifiés à l'aide de l'attribut `name` des éléments `<include>` et `<exclude>` utilisent les séparateurs de chemin et les conventions appropriés à la plateforme sur laquelle l'agent s'exécute. Si vous spécifiez des chemins de fichier relatifs, les chemins sont résolus par rapport à la propriété `transferRoot` de l'agent.

Lors de la spécification d'une restriction de file d'attente, la syntaxe `QUEUE@QUEUEMANAGER` est prise en charge, avec les règles suivantes:

- Si le caractère at (@) est manquant dans l'entrée, le modèle est traité comme un nom de file d'attente accessible sur n'importe quel gestionnaire de files d'attente. Par exemple, si le modèle est `name`, il est traité de la même manière que `name@**`.
- Si le caractère arobase (@) est le premier caractère de l'entrée, le modèle est traité comme un nom de gestionnaire de files d'attente et toutes les files d'attente du gestionnaire de files d'attente sont accessibles. Par exemple, si le modèle est `@name`, il est traité de la même manière que `**@name`.

Les caractères génériques suivants ont une signification spéciale lorsque vous les spécifiez dans le cadre de l'attribut `name` des éléments `<include>` et `<exclude>` :

Un astérisque unique correspond à zéro ou plusieurs caractères dans un nom de répertoire ou dans un qualificateur d'un nom de fichier ou d'un nom de file d'attente .

?

Un point d'interrogation correspond exactement à un caractère dans un nom de répertoire ou dans un qualificateur d'un nom de fichier ou d'un nom de file d'attente .

Deux astérisques correspondent à zéro ou plusieurs noms de répertoire, ou à zéro ou plusieurs qualificatifs dans un nom de fichier ou un nom de file d'attente . En outre, les chemins qui se terminent par un séparateur de chemin ont un "***" implicite ajouté à la fin du chemin. Ainsi, `/home/user/` est identique à `/home/user/**`.

Exemple :

- `/**/test/**` correspond à tout fichier dont le chemin contient un répertoire `test`
- `/test/file?` correspond à tout fichier du répertoire `/test` qui commence par la chaîne `file` suivie d'un caractère unique
- `c:\test*.txt` correspond à tout fichier du répertoire `c:\test` avec une extension `.txt`
- `c:\test***.txt` correspond à n'importe quel fichier du répertoire `c:\test` ou à l'un de ses sous-répertoires dont l'extension est `.txt`

- `z/OS` `/// 'TEST.*.DATA'` correspond à tout fichier dont le premier qualificateur est TEST, dont le second est un qualificateur et dont le troisième est DATA.
- `*@QM1` correspond à toute file d'attente du gestionnaire de files d'attente QM1 comportant un qualificateur unique.
- `TEST.*.QUEUE@QM1` correspond à n'importe quelle file d'attente du gestionnaire de files d'attente QM1 qui possède le premier qualificateur TEST, un deuxième qualificateur et un troisième qualificateur QUEUE.
- `**@QM1` correspond à n'importe quelle file d'attente du gestionnaire de files d'attente QM1.

Liens symboliques

Vous devez résoudre complètement tous les liens symboliques que vous utilisez dans les chemins de fichier du fichier `UserSandboxes.xml` en spécifiant des liens fixes dans les éléments `<include>` et `<exclude>`. Par exemple, si vous disposez d'un lien symbolique dans lequel `/var` est mappé à `/SYSTEM/var`, vous devez spécifier ce chemin en tant que `<tns:include name="/SYSTEM/var"/>`, sinon le transfert prévu échoue avec une erreur de sécurité du bac à sable de l'utilisateur.

Exemple

Cet exemple montre comment autoriser l'utilisateur avec le nom d'utilisateur MQMD `guest` à transférer un fichier depuis le répertoire `/home/user/public` ou l'un de ses sous-répertoires sur le système où l'agent `AGENT_JUPITER` est en cours d'exécution, en ajoutant l'élément `<sandbox>` suivant au fichier `UserSandboxes.xml` dans le répertoire de configuration d'`AGENT_JUPITER`:

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:userSandboxes
  xmlns:tns="http://wmqfte.ibm.com/UserSandboxes"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/UserSandboxes UserSandboxes.xsd">
  <tns:agent>
    <tns:sandbox user="guest">
      <tns:read>
        <tns:include name="/home/user/public/**"/>
      </tns:read>
    </tns:sandbox>
  </tns:agent>
</tns:userSandboxes>
```

Exemple

Cet exemple montre comment autoriser tout utilisateur avec le nom d'utilisateur MQMD `account` suivi d'un chiffre unique, par exemple `account4`, à effectuer les actions suivantes:

- Transférez tout fichier à partir du répertoire `/home/account` ou de l'un de ses sous-répertoires, à l'exception du répertoire `/home/account/private` sur le système où l'agent `AGENT_SATURN` est en cours d'exécution
- Transférez tout fichier dans le répertoire `/home/account/output` ou dans l'un de ses sous-répertoires sur le système où l'agent `AGENT_SATURN` est en cours d'exécution.
- Lire les messages des files d'attente du gestionnaire de files d'attente local en commençant par le préfixe `ACCOUNT.`, sauf s'il commence par `ACCOUNT.PRIVATE.` (c'est-à-dire avec `PRIVATE` au deuxième niveau).
- Transférez les données dans les files d'attente en commençant par le préfixe `ACCOUNT.OUTPUT.` sur n'importe quel gestionnaire de files d'attente.

Pour permettre à un utilisateur doté du nom d'utilisateur MQMD `account` d'effectuer ces actions, ajoutez l'élément `<sandbox>` suivant au fichier `UserSandboxes.xml`, dans le répertoire de configuration d'`AGENT_SATURN`:

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:userSandboxes
```

```

xmlns:tns="http://wmqfte.ibm.com/UserSandboxes"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://wmqfte.ibm.com/UserSandboxes UserSandboxes.xsd">
<tns:agent>
  <tns:sandbox user="account[0-9]" userPattern="regex">
    <tns:read>
      <tns:include name="/home/account/**"/>
      <tns:include name="ACCOUNT.**" type="queue"/>
      <tns:exclude name="ACCOUNT.PRIVATE.**" type="queue"/>
      <tns:exclude name="/home/account/private/**"/>
    </tns:read>
    <tns:write>
      <tns:include name="/home/account/output/**"/>
      <tns:include name="ACCOUNT.OUTPUT.**" type="queue"/>
    </tns:write>
  </tns:sandbox>
</tns:agent>
</tns:userSandboxes>

```

Référence associée

«Vérifications supplémentaires pour les transferts de caractères génériques», à la page 51

Depuis IBM MQ 9.0.1 dans Continuous Delivery et IBM MQ 9.0.0 Fix Pack 1 dans Long Term Support, si un agent a été configuré avec un bac à sable d'utilisateur ou d'agent afin de restreindre les emplacements vers et depuis lesquels l'agent peut transférer des fichiers, vous pouvez spécifier que des vérifications supplémentaires doivent être effectuées sur les transferts de caractères génériques pour cet agent.

«Utilisation des bacs à sable utilisateur MFT», à la page 48

Vous pouvez restreindre la zone du système de fichiers dans laquelle les fichiers peuvent être transférés en fonction du nom d'utilisateur MQMD qui demande le transfert.

«Le fichier MFT agent.properties», à la page 564

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.

V 9.0.1 **V 9.0.0.1** Vérifications supplémentaires pour les transferts de caractères génériques

Depuis IBM MQ 9.0.1 dans Continuous Delivery et IBM MQ 9.0.0 Fix Pack 1 dans Long Term Support, si un agent a été configuré avec un bac à sable d'utilisateur ou d'agent afin de restreindre les emplacements vers et depuis lesquels l'agent peut transférer des fichiers, vous pouvez spécifier que des vérifications supplémentaires doivent être effectuées sur les transferts de caractères génériques pour cet agent.

Propriété additionalWildcardSandboxChecking

Pour activer une vérification supplémentaire des transferts de caractères génériques, ajoutez la propriété suivante au fichier `agent.properties` de l'agent que vous souhaitez vérifier.

```
additionalWildcardSandboxChecking=true
```

Lorsque cette propriété est définie sur `true` et que l'agent effectue une demande de transfert qui tente de lire un emplacement qui se trouve en dehors du bac à sable défini pour la correspondance de fichier du caractère générique, le transfert échoue. S'il existe plusieurs transferts dans une même demande de transfert et que l'une de ces demandes échoue en raison d'une tentative de lecture d'un emplacement en dehors du bac à sable, le transfert complet échoue. Si la vérification échoue, la cause de l'échec est indiquée dans un message d'erreur.

Si la propriété `additionalWildcardSandboxChecking` est omise du fichier `agent.properties` d'un agent ou est définie sur `false`, aucune vérification supplémentaire n'est effectuée sur les transferts de caractères génériques pour cet agent.

Messages d'erreur pour la vérification des caractères génériques

Depuis IBM MQ 9.0.1 dans Continuous Delivery et IBM MQ 9.0.0 Fix Pack 1 dans Long Term Support, les messages qui sont signalés lorsqu'une demande de transfert générique est effectuée vers un emplacement en dehors d'un emplacement de bac à sable configuré ont été modifiés.

Le message suivant apparaît lorsqu'un chemin de fichier générique dans une demande de transfert se trouve en dehors du bac à sable restreint:

BFGSS0077E: La tentative de lecture du chemin d'accès au fichier *chemin* a été refusée. Le chemin d'accès au fichier se trouve hors du bac à sable de transfert restreint.

Le message suivant se produit lorsqu'un transfert au sein d'une demande de transfert multiple contient une demande de transfert générique dans laquelle le chemin se trouve en dehors du bac à sable restreint:

BFGSS0078E: La tentative de lecture du chemin d'accès au fichier: *chemin* a été ignorée car un autre transfert dans le transfert géré, tentative de lecture en dehors du bac à sable de transfert restreint.

Le message suivant s'affiche lorsqu'un fichier se trouve en dehors du bac à sable restreint:

BFGSS0079E: La tentative de lecture du fichier *chemin d'accès au fichier* a été refusée. Le fichier se trouve en dehors du bac à sable de transfert restreint.

Le message suivant se produit dans une demande de transfert multiple où une autre demande de transfert générique a entraîné la non-prise en compte de cette demande:

BFGSS0080E: La tentative de lecture du fichier: *chemin d'accès au fichier* a été ignorée car un autre transfert dans le transfert géré, tentative de lecture en dehors du bac à sable de transfert restreint.

Dans le cas de transferts de fichiers uniques qui n'incluent pas de caractères génériques, le message qui est signalé lorsque le transfert implique un fichier situé en dehors du bac à sable est inchangé par rapport aux éditions précédentes:

Echec avec BFGI00056E: La tentative de lecture du fichier "FILE" a été refusée. Le fichier se trouve en dehors du bac à sable de transfert restreint.

Référence associée

«Utilisation des bacs à sable utilisateur MFT», à la page 48

Vous pouvez restreindre la zone du système de fichiers dans laquelle les fichiers peuvent être transférés en fonction du nom d'utilisateur MQMD qui demande le transfert.

«Utilisation des bacs à sable d'agent MFT», à la page 46

Pour ajouter un niveau de sécurité supplémentaire à Managed File Transfer, vous pouvez restreindre la zone d'un système de fichiers à laquelle un agent peut accéder.

«Le fichier MFT `agent.properties`», à la page 564

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.

Configuration du chiffrement SSL ou TLS pour MFT

Vous pouvez utiliser SSL ou TLS avec IBM MQ Managed File Transfer pour sécuriser les communications entre les agents et leurs gestionnaires de files d'attente d'agent, les commandes et les gestionnaires de files d'attente auxquels ils se connectent, ainsi que les différentes connexions entre les gestionnaires de files d'attente et les gestionnaires de files d'attente dans votre topologie.

Avant de commencer

Vous pouvez utiliser le chiffrement SSL ou TLS pour chiffrer les messages qui transitent par une topologie IBM MQ Managed File Transfer. Ces gestionnaires sont les suivants :

- Messages transmis entre un agent et son gestionnaire de files d'attente d'agent.
- Messages des commandes et des gestionnaires de files d'attente auxquels elles se connectent.
- Messages internes qui circulent entre les gestionnaires de files d'attente d'agent, les gestionnaires de files d'attente de commandes et le gestionnaire de files d'attente de coordination dans la topologie.

Pourquoi et quand exécuter cette tâche

Pour des informations générales sur l'utilisation de SSL avec IBM MQ, voir [Utilisation de SSL ou TLS](#). En termes IBM MQ , Managed File Transfer est une application client Java standard.

Pour utiliser SSL avec Managed File Transfer, procédez comme suit:

Procédure

1. Créez un fichier de clés certifiées et éventuellement un fichier de clés (ces fichiers peuvent être identiques). Si vous n'avez pas besoin de l'authentification client (c'est-à-dire, SSLCAUTH=OPTIONAL sur les canaux), vous n'avez pas besoin de fournir un magasin de clés. Vous avez besoin d'un magasin de clés de confiance uniquement pour authentifier le certificat du gestionnaire de files d'attente.

L'algorithme de clé utilisé pour créer des certificats pour le magasin de clés de confiance et les magasins de clés doit être RSA pour fonctionner avec IBM MQ.

2. Configurez votre gestionnaire de files d'attente IBM MQ pour utiliser SSL.
Pour plus d'informations sur la configuration d'un gestionnaire de files d'attente pour utiliser SSL avec IBM MQ Explorer , par exemple, voir [Configuration de SSL sur les gestionnaires de files d'attente](#).
3. Sauvegardez le fichier de clés certifiées et le fichier de clés (si vous en avez un) dans un emplacement approprié. Un emplacement suggéré est le répertoire `config_directory/coordination_qmgr/agents/agent_name` .
4. Définissez les propriétés SSL requises pour chaque gestionnaire de files d'attente SSL dans le fichier de propriétés Managed File Transfer approprié. Chaque ensemble de propriétés fait référence à un gestionnaire de files d'attente distinct (agent, coordination et commande), bien qu'un gestionnaire de files d'attente puisse exécuter deux ou plusieurs de ces rôles.

L'une des propriétés **CipherSpec** ou **CipherSuite** est requise, sinon le client tente de se connecter sans SSL. Les propriétés **CipherSpec** ou **CipherSuite** sont fournies en raison des différences de terminologie entre IBM MQ et Java. Managed File Transfer accepte l'une ou l'autre propriété et effectue la conversion nécessaire. Vous n'avez donc pas besoin de définir les deux propriétés. Si vous spécifiez à la fois les propriétés **CipherSpec** ou **CipherSuite** , **CipherSpec** est prioritaire.

La propriété **PeerName** est facultative. Vous pouvez définir la propriété sur le nom distinctif du gestionnaire de files d'attente auquel vous souhaitez vous connecter. Managed File Transfer rejette les connexions à un serveur SSL incorrect avec un nom distinctif qui ne correspond pas.

Définissez les propriétés **SslTrustStore** et **SslKeyStore** sur des noms de fichier qui pointent vers les fichiers de clés certifiées et les fichiers de clés. Si vous configurez ces propriétés pour un agent déjà en cours d'exécution, arrêtez et redémarrez l'agent pour qu'il se reconnecte en mode SSL.

Les fichiers de propriétés contiennent des mots de passe en texte en clair. Il est donc judicieux de définir les droits d'accès appropriés au système de fichiers.

Pour plus d'informations sur les propriétés SSL, voir «Propriétés SSL pour MFT», à la page 622.

5. Si un gestionnaire de files d'attente d'agent utilise SSL, vous ne pouvez pas fournir les détails nécessaires lors de la création de l'agent. Procédez comme suit pour créer l'agent:
 - a) Créez l'agent à l'aide de la commande **fteCreateAgent** . Vous recevez un avertissement indiquant que vous ne parvenez pas à publier l'existence de l'agent dans le gestionnaire de file d'attente de coordination.
 - b) Editez le fichier `agent.properties` créé à l'étape précédente pour ajouter les informations SSL. Lorsque l'agent est correctement démarré, une nouvelle tentative de publication est effectuée.
6. Si des agents ou des instances de IBM MQ Explorer sont en cours d'exécution alors que les propriétés SSL du fichier `agent.properties` ou du fichier `coordination.properties` sont modifiées, vous devez redémarrer l'agent ou IBM MQ Explorer.

Référence associée

«Le fichier MFT `agent.properties`», à la page 564

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.

Connexion à un gestionnaire de files d'attente en mode client avec authentification de canal

IBM WebSphere MQ 7.1 a introduit des enregistrements d'authentification de canal pour contrôler plus précisément l'accès au niveau du canal. Ce changement de comportement signifie que par défaut, les gestionnaires de files d'attente IBM WebSphere MQ 7.1 ou version ultérieure nouvellement créés rejettent les connexions client à partir du composant Managed File Transfer .

Pour plus d'informations sur l'authentification de canal, voir [Enregistrements d'authentification de canal](#).

Si la configuration de l'authentification de canal pour le SVRCONN utilisé par Managed File Transfer spécifie un ID MCAUSER non privilégié, vous devez accorder des enregistrements de droits d'accès spécifiques pour le gestionnaire de files d'attente, les files d'attente et les rubriques, afin de permettre au Managed File Transfer Agent et aux commandes de fonctionner correctement. Utilisez la commande `MQSC SET CHLAUTH` ou la commande `PCF Set Channel Authentication Record` pour créer, modifier ou supprimer des enregistrements d'authentification de canal. Pour tous les agents Managed File Transfer que vous souhaitez connecter au gestionnaire de files d'attente IBM WebSphere MQ 7.1 ou version ultérieure, vous pouvez soit configurer un ID MCAUSER à utiliser pour tous vos agents, soit configurer un ID MCAUSER distinct pour chaque agent.

Accordez à chaque ID MCAUSER les droits suivants:

- Enregistrements de droits d'accès requis pour le gestionnaire de files d'attente:
 - connect
 - setid
 - inq
- Enregistrements de droits d'accès requis pour les files d'attente.

Pour toutes les files d'attente spécifiques à l'agent, c'est-à-dire les noms de file d'attente qui se terminent par `nom_agent` dans la liste suivante, vous devez créer ces enregistrements de droits d'accès aux files d'attente pour chaque agent que vous souhaitez connecter au gestionnaire de files d'attente IBM WebSphere MQ 7.1 ou version ultérieure à l'aide d'une connexion client.

- put, get, dsp (SYSTEM.DEFAULT.MODEL.QUEUE)
 - put, get, setid, browse (SYSTEM.FTE.COMMAND.*nom_agent*)
 - put, get (SYSTEM.FTE.DATA.*nom_agent*)
 - put, get (SYSTEM.FTE.REPLY.*nom_agent*)
 - put, get, inq, browse (SYSTEM.FTE.STATE.*nom_agent*)
 - put, get, browse (SYSTEM.FTE.EVENT.*nom_agent*)
 - put, get (SYSTEM.FTE)
- Enregistrements de droits d'accès requis pour les rubriques:
 - sub, pub (SYSTEM.FTE)
 - Enregistrements de droits d'accès requis pour les transferts de fichiers.

Si vous disposez d'ID MCAUSER distincts pour l'agent source et l'agent de destination, créez les enregistrements de droits d'accès dans les files d'attente des agents à la fois à la source et à la destination.

Par exemple, si l'ID MCAUSER de l'agent source est **user1** et l'ID MCAUSER de l'agent cible est **user2**, définissez les droits suivants pour les utilisateurs de l'agent:

Utilisateur d'agent	File d'attente	droits requis
user1	SYSTEME SYSTEM.FTE.DATA. <i>nom_agent_destination</i>	put


```

csBXUniW4A3UzZnCRsv3MB8GA1UdIwQYMBaAFDXY8rjmj41Vz5+FVAoQb++cns+B4
MA0GCSqSb3DQEBBQUAA4GBAfc7k1Xa4pGKYgwhcxKpE3ZF6FNWY4vBXS216/ja
8h/v18+iv010CL8t0ZOKSU95fyZLz0PKnCH7v+ITFSE3CIIEk9D1z2U6W091ICwn
17PL72Tdfal3kabwHYVf17IVcuL+VZsZ3HjLggP2qH09ZuJPspeT9+AxFVMLiaAb
8eHw
-----END CERTIFICATE-----
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,64A02DA15B6B6EF9

57kqxL0J/gRU0IQ6hVK2YN13B4E1jAi1gSme0I5ZpEIG8CHXISKB7/0cke2FTqsV
1vI99QyCxsDwoMnt5fj51v7aPmVeS60b0m+U1Gre8B/Ze18JVj204K2Uh72rDCXE
5e6eFxsDUM207sQDy20euBVELJtM2k0kL1RodoQOS1U3XQNgJw/t3ZIx5hPXWEQT
rjRQ064BEhb+PzzxPF8uwzZ9IzUK9BJ/UUnqC60dBR87IeA4pnJD1Jvb2ML7EN9Z
5Y+50hTKI80GvBvWX04fHyvIX5as1whBoArXIS1AtNTtPv0aP1zyIAeZ60Cvo/
Sfo+A2UhmTEJe0JaZG2XZ3H495fAw/EHmjehzIACwukQ9nSIEtgu4A1+CV64RJED
aYBCM8UjaAkbZDH5gn7+eBov0ssXAXWDyJBVhU0jXjvAj/e1h+kcsF1hax5D//AI
66nRMZzboSxNqkjcVd8wfdWp+bEjDzUaaaTJTS7LIFeLLw7eJ8MNAkMGicDkycL0
EPBU9X5QnHKLK0fYHN/1WgUk8qt3UyTFXXfzTXGF3EbsWbBupkT5e5+1YcX80VZ6
sHFPN1HlucNy/riUcBy9iviVeodX8Iom0chSy05DK18bwZNjYtUP+CtYHNFU5BaD
I+1uU0AeJ+wjQYKT1WaeIGZ3VxuNITJu18y5qDTXXfX7vXm50owXa6U5+AYuGUMg
/itPZmUmNtHjTk7ghT6i1IQ0aBowXXKJB1Mmq/6BQXN2IhkD9ys2qrVM1hd15nAf
egmdiG501oLnBRqWbFR+DykpAhK4SaDi2F52Uxovw3Lhiw8dQP7lZQ==
-----END RSA PRIVATE KEY-----

```

7. Démarrez l'outil d'administration Secure +.

- Sur les systèmes Linux ou UNIX , exécutez la commande **spadmin.sh**.
- Sur les systèmes Windows , cliquez sur **Démarrer > Programmes > Sterling Commerce Connect:Direct > CD Secure + Admin Tool**

L'outil d'administration CD Secure + démarre.

8. Dans l'outil d'administration CD Secure +, cliquez deux fois sur **.Ligne** locale pour éditer les paramètres SSL ou TLS principaux.

- Sélectionnez **Activer le protocole SSL** ou **Activer le protocole TLS**, selon le protocole que vous utilisez.
- Sélectionnez **Désactiver la substitution**.
- Sélectionnez au moins une suite de chiffrement.
- Si vous souhaitez une authentification bidirectionnelle, remplacez la valeur de **Activer l'authentification client** par Yes.
- Dans la zone **Certificat racine accrédité** , entrez le chemin d'accès au fichier de certificat public de votre autorité de certification, /test/ssl/certs/CACert.
- Dans la zone **Fichier de certificat de clé** , entrez le chemin d'accès au fichier que vous avez créé, /test/ssl/cd/keyCertFile/node_name.txt.

9. Cliquez deux fois sur le **.Ligne** du client pour éditer les paramètres SSL ou TLS principaux.

- Sélectionnez **Activer le protocole SSL** ou **Activer le protocole TLS**, selon le protocole que vous utilisez.
- Sélectionnez **Désactiver la substitution**.

Pour l'agent de pont Connect:Direct , procédez comme suit:

10. Créez un magasin de clés de confiance. Pour ce faire, vous pouvez créer une clé factice, puis la supprimer.

Vous pouvez utiliser les commandes suivantes:

```
keytool -genkey -alias dummy -keystore /test/ssl/fte/stores/truststore.jks
```

```
keytool -delete -alias dummy -keystore /test/ssl/fte/stores/truststore.jks
```

11. Importez le certificat public de l'autorité de certification dans le magasin de clés de confiance.

Vous pouvez utiliser la commande suivante :

```
keytool -import -trustcacerts -alias myCA
        -file /test/ssl/certs/CAcert
        -keystore /test/ssl/fte/stores/truststore.jks
```

12. Editez le fichier de propriétés de l'agent de pont Connect:Direct .

Incluez les lignes suivantes n'importe où dans le fichier:

```
cdNodeProtocol=protocol
cdNodeTruststore=/test/ssl/fte/stores/truststore.jks
cdNodeTruststorePassword=password
```

Dans l'exemple de cette étape, *protocol* est le protocole que vous utilisez, SSL ou TLS, et *password* est le mot de passe que vous avez spécifié lors de la création du magasin de clés de confiance.

13. Si vous souhaitez une authentification bidirectionnelle, créez une clé et un certificat pour l'agent de pont Connect:Direct .

- a) Créez un magasin de clés et une clé.

Vous pouvez utiliser la commande suivante :

```
keytool -genkey -keyalg RSA -alias agent_name
        -keystore /test/ssl/fte/stores/keystore.jks
        -storepass password -validity 365
```

- b) Générez une demande de signature.

Vous pouvez utiliser la commande suivante :

```
keytool -certreq -v -alias agent_name
        -keystore /test/ssl/fte/stores/keystore.jks -storepass password
        -file /test/ssl/fte/requests/agent_name.request
```

- c) Importez le certificat que vous avez reçu à l'étape précédente dans le magasin de clés. Le certificat doit être au format x.509 .

Vous pouvez utiliser la commande suivante :

```
keytool -import -keystore /test/ssl/fte/stores/keystore.jks
        -storepass password -file certificate_file_path
```

- d) Editez le fichier de propriétés de l'agent de pont Connect:Direct .

Incluez les lignes suivantes n'importe où dans le fichier:

```
cdNodeKeystore=/test/ssl/fte/stores/keystore.jks
cdNodeKeystorePassword=password
```

Dans l'exemple de cette étape, *password* est le mot de passe que vous avez spécifié lors de la création du magasin de clés.

Tâches associées

«Configuration du pont Connect:Direct», à la page 146

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 .

configurationMFT

Vous pouvez configurer les fonctions d' Managed File Transfer après l'installation.

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

[«MFT Reprise et redémarrage», à la page 252](#)

Si votre agent ou votre gestionnaire de files d'attente n'est pas disponible pour une raison quelconque, par exemple en raison d'une panne d'alimentation ou de réseau, Managed File Transfer effectue la récupération suivante dans les scénarios suivants:

[«Conseils et astuces pour l'utilisation d' MFT», à la page 279](#)

Voici quelques suggestions pour vous aider à utiliser au mieux Managed File Transfer.

[«Administration d'MFT», à la page 157](#)

Utilisez les commandes Managed File Transfer pour administrer Managed File Transfer. Vous pouvez également utiliser le IBM MQ Explorer pour certaines des tâches d'administration.

Tâches associées

[«Surveillance des ressources MFT», à la page 174](#)

Vous pouvez surveiller les ressources Managed File Transfer ; par exemple, une file d'attente ou un répertoire. Lorsqu'une condition est satisfaite sur cette ressource, le moniteur de ressources démarre une tâche, telle qu'un transfert de fichier. Vous pouvez créer un moniteur de ressources à l'aide de la commande **fteCreateMonitor** ou de la vue **Moniteurs** dans le plug-in Managed File Transfer pour IBM MQ Explorer.

[«Configuration de MQMFTCredentials.xml sur plusieurs plateformes», à la page 43](#)

Si Managed File Transfer (MFT) est configuré avec la sécurité activée, l'authentification de connexion 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. De même, les consigneurs MFT peuvent être tenus de spécifier un ID utilisateur et un mot de passe lors de la connexion à une base de données. Ces données d'identification peuvent être stockées dans le fichier de données d'identification MFT .

Référence associée

[«Commandes MFT», à la page 383](#)

Toutes les commandes Managed File Transfer sont répertoriées avec des liens vers leurs descriptions détaillées.

[«Sécurisation de MFT», à la page 41](#)

Juste après l'installation et si vous n'avez apporté aucune modification, Managed File Transfer présente un niveau de sécurité pouvant être adapté à des fins de test ou d'évaluation dans un environnement protégé. Toutefois, dans un environnement de production, vous devez envisager de contrôler de façon appropriée les utilisateurs pouvant démarrer des opérations de transfert de fichier et lire et écrire les fichiers transférés, et déterminer comment protéger l'intégrité des fichiers.

[«Le fichier MFT agent.properties», à la page 564](#)

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.

[«Utilisation d' Apache Ant avec MFT», à la page 258](#)

Managed File Transfer fournit des tâches que vous pouvez utiliser pour intégrer la fonction de transfert de fichier dans l'outil Ant Apache .

[«traitement des incidentsMFT», à la page 277](#)

Utilisez les informations de référence suivantes pour vous aider à diagnostiquer les erreurs dans Managed File Transfer:

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` spécifie 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 de l'ensemble d'options de configuration correspond au nom du gestionnaire de files d'attente de coordination associé. Pour plus d'informations sur le fichier `installation.properties` , voir [«Le fichier MFT installation.properties»](#), à la page 549.

coordination.properties

Le fichier `coordination.properties` spécifie les détails de connexion au gestionnaire de files 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»](#), à la page 553.

command.properties

Le fichier `command.properties` indique le gestionnaire de files d'attente de commandes auquel vous devez vous connecter lorsque vous émettez des commandes et les informations dont Managed File Transfer a besoin pour contacter ce gestionnaire de files d'attente. Pour plus d'informations sur le fichier `command.properties`, voir [«Le fichier MFT `command.properties`»](#), à la page 559.

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`»](#), à la page 564.

logger.properties

Le fichier `logger.properties` spécifie les propriétés de configuration des consignateurs. Pour plus d'informations sur le fichier `logger.properties`, voir [«Propriétés de configuration du consignateur MFT»](#), à la page 120.

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.

z/OS

Options de configuration MFT sous z/OS

Les options de configuration Managed File Transfer sous z/OS sont identiques à celles des plateformes réparties.

Pour plus d'informations sur les options de configuration sous [Multiplateformes](#), voir [«Options de configuration de MFT sur Multiplatforms»](#), à la page 58.

Sous z/OS, l'emplacement de configuration est défini par la variable d'environnement `BFG_DATA`. Si une configuration n'existe pas déjà dans le répertoire UNIX System Services référencé par `BFG_DATA`, le script `JCL BFGCUSTOM` d'un fichier de bibliothèque PDSE de commande MFT génère les travaux requis pour créer la configuration. La configuration est ensuite créée lorsque vous exécutez ces travaux générés. La création de la configuration repose sur le fait que `BFG_DATA` fait référence à un répertoire existant qui est accessible.

Vous pouvez également créer et gérer une configuration à l'aide des mêmes commandes **fte** disponibles sur [Multiplatforms](#) et z/OS. Pour la liste des commandes **fte**, voir [«Commandes MFT»](#), à la page 383.

V 9.0.1

Configuration de l'Redistributable Managed File Transfer Agent

Configurez le Redistributable Managed File Transfer Agent éventuellement fourni pour vous connecter à une infrastructure IBM MQ existante et permettre aux utilisateurs de transférer des fichiers sans installer IBM MQ afin d'obtenir la fonctionnalité Managed File Transfer.

Avant de commencer

Redistributable Managed File Transfer Agent fournit les fonctionnalités de Managed File Transfer avec les exceptions suivantes:

- La connexion en mode liaisons 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.
- Les commandes suivantes ne sont pas incluses:
 - `fteCreateCDAgent.cmd`
 - `fteCreateLogger.cmd`

- fteDeleteLogger.cmd
- fteMigrateLogger.cmd
- fteSetLoggerTraceLevel.cmd
- fteShowLoggerDetails.cmd
- fteStartLogger.cmd
- fteStopLogger.cmd

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 suivantes sur votre système pour utiliser Redistributable Managed File Transfer Agent:

- Microsoft Visual C++ Redistributable 2008
- Microsoft Visual C++ Redistributable 2012

Ces bibliothèques sont disponibles dans Microsoft. Voir [Les derniers téléchargements Visual C++ pris en charge](#).

Pourquoi et quand exécuter cette tâche

Vous souhaitez 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 pouvoir le faire. Procédez comme suit pour télécharger et extraire le fichier Redistributable Managed File Transfer Agent:

Procédure

1. Téléchargez le package Redistributable Managed File Transfer Agent à partir d' [IBM Fix Central](#).

a) Choisissez le package correspondant à votre système d'exploitation:

- **Windows** 9.0.1.0-IBM-MQFA-Redist-Win64
- **Linux** 9.0.1.0-IBM-MQFA-Redist-LinuxX64
- **Linux** 9.0.1.0-IBM-MQFA-Redist-LinuxS390X
- **Linux** 9.0.1.0-IBM-MQFA-Redist-LinuxPPC64LE

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** Utilisez les outils Windows Explorer pour les extraire.
- **Linux** Extraire et décompresser comme suit:

```
gunzip 9.0.1.0-IBM-MQFA-Redist-LinuxX64.tar.gz
```

puis

```
tar xvf 9.0.1.0-IBM-MQFA-Redist-LinuxX64.tar
```

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** > **Linux** mqft: contient les répertoires ant et lib requis pour la prise en charge d'Ant et pour la prise en charge de la fonction MFT de base
- **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 l'agent MFT . Pour les étapes suivantes, voir la rubrique suivante: [«Création de la configuration initiale»](#), à la page 62.

Référence associée

«Erreurs possibles lors de la configuration de l'agent MFT redistribuable», à la page 328
Messages d'erreur lors de la configuration du Redistributable Managed File Transfer Agent

Information associée

Vidéo: [Comment installer et configurer Redistributable Managed File Transfer Agent \(YouTube\)](#)

Création de la configuration initiale

Vous pouvez configurer un Managed File Transfer Agent pour qu'il se connecte à une configuration IBM MQ existante.

Avant de commencer

V 9.0.1 Veuillez à télécharger et à extraire le contenu du package Redistributable Managed File Transfer Agent . Pour plus d'informations, voir [«Configuration de l'Redistributable Managed File Transfer Agent»](#), à la page 60.

Pourquoi et quand exécuter cette tâche

Créez l'environnement pour pouvoir utiliser les commandes permettant de configurer la connectivité avec le gestionnaire de files d'attente qui s'exécute sur le serveur IBM MQ .

Procédure

1. Créez l'environnement pour Redistributable Managed File Transfer Agent.

Lorsque vous exécutez la commande **fteCreateEnvironment** , le répertoire de données MFT contenant les informations de configuration des agents MFT est créé. Vérifiez que vous vous trouvez dans le répertoire bin qui a été créé lorsque vous avez extrait le composant redistribuable de l'agent MFT téléchargé. Exécutez ensuite la commande suivante :

- **Windows**

```
fteCreateEnvironment.cmd -d datapath location
```

- **Linux**

```
./fteCreateEnvironment -d datapath location
```

Cette commande accepte un paramètre facultatif:

-d

Emplacement du chemin de données dans lequel la configuration MFT est créée, stockée et gérée. Si l'utilisateur exécute **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 doit s'exécuter en tant que service Windows, vous devez définir la variable d'environnement **BFG_DATA** dans l'environnement système pour que le service fonctionne.

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 fonctionnant en mode client, vous devez 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 d'informations sur l'utilisation de la commande **fteSetupCoordination** , voir «[fteSetupCoordination](#)», à la page 527. 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 99.

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 d'informations sur l'utilisation de la commande **fteSetupCommands** , voir «[Commandes fteSetup: créez le fichier MFT command.properties](#)», à la page 524.

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 «[Agent fteCreate\(créer un agent MFT \)](#)», à la page 399.

Dans les étapes «2», à la page 63 et «3», à la page 63 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 d'informations sur l'utilisation de la commande **fteListAgents**, voir «Agents fteList: répertorie les agents MFT pour un gestionnaire de files d'attente de coordination», à la page 484.

Concepts associés

«configurationMFT», à la page 57

Vous pouvez configurer les fonctions d' Managed File Transfer après l'installation.

«Options de configuration de MFT sur Multiplatforms», à la page 58

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», à la page 447

La commande **fteCreateTransfer** crée et démarre un nouveau transfert de fichier à partir de la ligne de commande. Cette commande peut démarrer un transfert de fichier immédiatement, planifier un transfert de fichier à une heure et une date ultérieures, répéter un transfert planifié une ou plusieurs fois et déclencher un transfert de fichiers sous certaines conditions.

Création d'un fichier de commandes d'agent ou de consignateur MFT

Vous pouvez créer un jeu de données PDSE de commandes à partir du jeu de données du modèle de commande Managed File Transfer pour un Managed File Transfer Agent ou Managed File Transfer Logger spécifique pour une coordination spécifique.

Pourquoi et quand exécuter cette tâche

Procédez comme suit :

Procédure

1. Effectuez une copie du fichier de bibliothèque PDSE du modèle de commande MFT SBFGCMD5.

 SBFGCMD5 doit être copié dans une nouvelle bibliothèque, par exemple *prefix.agent.JCL_*. Vous pouvez utiliser une version mise à jour du membre SBFGCMD5 (BFGCOPY) avec les remplacements suivants:

- Remplacez *++supplied-library++* par le nom qualifié complet du fichier PDSE SBFGCMD5.
 -  Remplacez *++service-library++* par le nom qualifié complet du nouveau fichier de bibliothèque PDSE de la commande MFT. *++service-library++* est le fichier de sortie du service d'agent ou de consignateur créé.
2. Pour le nouvel ensemble de données de la bibliothèque PDSE de commandes MFT, éditez le membre BFGCUSTOM, qui est un script JCL permettant de personnaliser les commandes de l'agent ou du consignateur. Chaque variable est spécifiée au format: *++nom de variable++*, que vous devez remplacer par sa valeur requise. Pour une description des différentes variables JCL, voir «z/OS variables JCL», à la page 78. L'instruction de définition de données BFGSTDIN définit les variables dans trois catégories: Variables, Propriétés et Environnement. Le format de l'instruction est le suivant:

```
[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
```

Les variables définissent l'ensemble des variables de configuration et d'environnement requises pour chaque commande.

Les propriétés définissent des substitutions pour les propriétés de configuration MFT . Vous pouvez ajouter des propriétés d'agent et de consignateur si nécessaire afin de personnaliser l'agent ou le consignateur pour votre environnement. Pour obtenir la liste de toutes les propriétés, voir [«Fichiers de propriétés de configuration»](#), à la page 91. Cette fonction permet d'éviter d'avoir à accéder aux fichiers de propriétés de configuration MFT , qui sont gérés en tant que fichiers UNIX System Services.

L'environnement définit toutes les variables d'environnement personnalisées supplémentaires requises.

3. Soumettez le travail BFGCUSTM pour le nouveau fichier de bibliothèque PDSE de la commande MFT . Ce travail génère l'ensemble de commandes JCL, en tant que nouveaux membres du PDSE, appropriées pour l'agent ou le consignateur. Pour la liste complète des commandes, voir [«Scripts JCL des commandes de l'agent et du consignateur z/OS»](#), à la page 83.

Le travail BFGCUSTM met à jour la bibliothèque contenant le JCL qui inclut une instruction de définition de données avec DISP=OLD. Vous devez quitter l'éditeur après la soumission pour permettre l'exécution du travail.

Examinez le journal du travail de sortie pour vérifier que le script JCL s'est exécuté correctement. S'il y a des échecs, corrigez-les et soumettez à nouveau le travail BFGCUSTM.

Le script JCL BFGCUSTM met également à jour les fichiers de propriétés de configuration UNIX System Services MFT selon les besoins pour conserver les fichiers à l'étape. Si la configuration définie par la propriété CoordinationQMgr n'existe pas, des messages d'avertissement sont générés et vous devez exécuter les travaux BFGCFR et BFGCMCR générés pour créer les fichiers de propriétés de configuration. Vous devez exécuter BFGAGCR pour un agent et BFGLGCRS pour une édition de consignateur. Si la configuration spécifiée existe déjà, elle est mise à jour avec les propriétés définies dans le script JCL BFTCUSTM.

Concepts associés

[«Options de configuration MFT sous z/OS»](#), à la page 60

Les options de configuration Managed File Transfer sous z/OS sont identiques à celles des plateformes réparties.

Tâches associées

[«Mise à jour d'un fichier de commandes d'agent ou de consignateur MFT existant sous z/OS»](#), à la page 78

Vous pouvez mettre à jour un fichier de bibliothèque PDSE de commande Managed File Transfer créé à partir du fichier de modèle de commande Managed File Transfer .

z/OS

Configuration de Managed File Transfer for z/OS

Managed File Transfer for z/OS nécessite une personnalisation pour permettre au composant de fonctionner correctement.

Pourquoi et quand exécuter cette tâche

Vous devez :

1. Edition d'un membre PDSE pour spécifier des données de configuration
2. Définissez le gestionnaire de file d'attente de coordination.
3. Définition du gestionnaire de files d'attente de commandes
4. Configurer un ou plusieurs agents
5. Facultatif: configurez une tâche de consignateur pour stocker des données dans Db2

La séquence des tâches que vous devez effectuer est détaillée dans les rubriques suivantes.

Concepts associés

«Vérification de la configuration de MFT», à la page 66

Vous devez vérifier la configuration de votre système avant de commencer.

Information associée

[Installation Managed File Transfer for z/OS](#)

Vérification de la configuration de MFT

Vous devez vérifier la configuration de votre système avant de commencer.

Managed File Transfer (MFT) requiert qu'un ou plusieurs gestionnaires de files d'attente agissent dans les rôles suivants pour chaque configuration MFT définie:

- Un gestionnaire de files d'attente de coordination, qui gère des informations sur le statut de chaque agent dans la configuration publiée dans une rubrique du coordinateur.
- Un ou plusieurs gestionnaires de files d'attente de commandes ou de connexions qui servent de point d'entrée au réseau IBM MQ pour les commandes MFT.
- Un ou plusieurs gestionnaires de files d'attente d'agent qui fournissent la communication entre un agent MFT et le réseau IBM MQ .

Chacun des rôles ci-dessus peut être exécuté par un gestionnaire de files d'attente distinct ou vous pouvez combiner les rôles de sorte que, dans la configuration la plus simple, tous les rôles soient exécutés par un seul gestionnaire de files d'attente.

Si vous ajoutez un gestionnaire de files d'attente z/OS à un environnement MFT existant, vous devez définir la connectivité entre le gestionnaire de files d'attente z/OS et les autres gestionnaires de files d'attente de la configuration. Pour ce faire, vous pouvez utiliser des files d'attente de transmission définies manuellement ou utiliser la mise en cluster.

Chaque agent MFT communique avec un seul gestionnaire de files d'attente. Si plusieurs agents communiquent avec le même gestionnaire de files d'attente, plusieurs files d'attente sont définies pour chaque agent:

- SYSTEME SYSTEM.FTE.COMMAND.*nom_agent*
- SYSTEME SYSTEM.FTE.DATA.*nom_agent*
- SYSTEME SYSTEM.FTE.REPLY.*nom_agent*
- SYSTEME SYSTEM.FTE.STATE.*nom_agent*
- SYSTEME SYSTEM.FTE.EVENT.*nom_agent*
- SYSTEM.FTE.AUTHAGT1.*nom_agent*
- SYSTEM.FTE.AUTHTRN1.*nom_agent*
- SYSTEM.FTE.AUTHOPS1.*nom_agent*
- SYSTEM.FTE.AUTHSCH1.*nom_agent*
- SYSTEM.FTE.AUTHMON1.*nom_agent*
- SYSTEM.FTE.AUTHADM1.*nom_agent*

Notez que vous pouvez définir des profils de sécurité génériques, dans lesquels vous utilisez un profil tel que SYSTEM.FTE.COMMAND.* , ou vous pouvez définir des profils spécifiques pour chaque agent.

Concepts associés

«Avant de commencer», à la page 66

La configuration Managed File Transfer (MFT) utilise des fichiers dans les fichiers UNIX System Services (USS) et PDSE.

Avant de commencer

La configuration Managed File Transfer (MFT) utilise des fichiers dans les fichiers UNIX System Services (USS) et PDSE.

La plupart de la configuration et de l'opération sont effectuées à l'aide de JCL provenant d'un ensemble de données partitionnées étendu (PDSE) et vous devez être familiarisé avec l'utilisation d'un environnement USS.

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 USS suivantes

Commande	Function
ls -ltr chemin	Répertorie les informations sur les fichiers dans le chemin d'accès
répertoire ls -ltrd	Répertorie des informations sur le répertoire spécifié plutôt que sur les fichiers du répertoire.
rechercher le chemin -name xxx	Recherchez le fichier nommé xxx dans le répertoire du chemin d'accès. xxx est sensible à la casse et peut être similaire à *zzz
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
Nom de fichier oedit	Editer un fichier dans OMVS
nom de fichier obrowse	Parcourir le nom de fichier

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
BFG_JAVA_HOME	/java/java71_bit64_GA/J7.1_64/	
BFG_GROUP_NAME	MQM	
LIBRARY	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.
TMPDIR	/tmp	Chemin USS accessible en lecture et en écriture pour les fichiers temporaires.
SERVICE_TYPE	AGENT ou LOGGER	
NOM	AGENT1	
BFG_PROD	/var/ibm/wmqmft	

Tableau 4. Paramètres requis pour le membre BFGCUSTM (suite)		
Nom	Exemple de données	Commentaires
BFG_DATA		Complétez le cas échéant
BFG_JVM_PROPERTIES		Complétez le cas échéant
QMGR	MQPV	
MQ_PATH	/mqm/V8R0M0	
MQ_LANG	E	
Db2_HLQ	SYS2.Db2.Version 10	
FTE_CONFIG		Utilisé dans la migration
CHEMIN_DONNÉES_D'ACCÈS		Utilisé dans la migration
DB_PROPS_PATH		Utilisé dans la migration
BFG_WTO	YES	Pour obtenir un message MFT sur le syslog.
ADMIN_JOB1		Carte de travail. Tous les travaux sont générés avec la même carte JCL.
CLASSE DE SORTIE	*	
JOBCARD1		Il s'agit de la carte de travail pour les tâches à exécution longue, les agents et les consignateurs.
Voie	bin:/usr/bin:/usr/sbin	
armELEMTYPE	Si ARM est utilisé, utilisez le paramètre ARM ELEMTYPE spécifié dans la règle ARM. Par exemple, armELEMTYPE= SYSBFGAG pour un agent ou armELEMTYPE= SYSBFGLG pour un consignateur. Si ARM n'est pas utilisé, mettez ce paramètre à blanc ; par exemple, armELEMTYPE=	
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=	
coordinationQMgr	MQPV	Configuration obligatoire

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 ne doivent pas être renseignés pour une connexion de liaisons sur la machine locale.

Concepts associés

«Eléments à vérifier», à la page 69

Vérifiez que vous disposez de suffisamment d'espace disque, d'un répertoire pour le stockage des données et que les fichiers requis existent.

«Edition du membre BFGCUSTM», à la page 74

Vous devez éditer le membre BFGCUSTM et entrer les valeurs des paramètres utilisés par votre entreprise avant d'exécuter le travail.

Eléments à vérifier

Vérifiez que vous disposez de suffisamment d'espace disque, d'un répertoire pour le stockage des données et que les fichiers requis existent.

Vérifiez que vous disposez de suffisamment d'espace disque

Vérifiez que vous disposez de suffisamment d'espace disque disponible sur le système de fichiers dans lequel vous allez stocker les fichiers spécifiques à la configuration.

Si une trace d'agent est activée, elle peut utiliser par défaut 100 Mo d'espace disque.

Les fichiers de configuration eux-mêmes sont petits, de quelques Ko seulement.

Si vous prévoyez d'utiliser deux agents et un consignateur, vous avez besoin d'au moins 300 Mo.

Vous pouvez utiliser la commande **df -k path**, où path est l'emplacement des fichiers spécifiques à l'installation. Indiquez l'espace disponible et l'espace total en Ko.

300 Mo correspond à 307 200 Ko, vous devez donc autoriser au moins 310 000 Ko

Création et vérification du répertoire de stockage des données Managed File Transfer

Vous avez besoin d'un répertoire pour stocker les données Managed File Transfer (MFT).

Vérifiez que vous disposez de suffisamment d'espace dans le système de fichiers **df -k /var**. Ce système de fichiers doit disposer d'au moins 310 000 Ko.

Si vous n'avez pas créé ce système de fichiers, utilisez la commande **mkdir**, par exemple **mkdir /var/mft**.

Affichez les droits des utilisateurs sur ce répertoire à l'aide de la commande **ls -ltrd /var/mft**.

Si le propriétaire ou le groupe n'est pas correct, utilisez la commande **chown owner:group /var/mft**.

Si les droits du groupe ne sont pas corrects, utilisez la commande suivante pour accorder au propriétaire et au groupe des droits de lecture, d'écriture et d'exécution. Notez que la commande suivante donne également à tous les utilisateurs les droits de lecture et d'exécution **chmod 775 /var/mft**.

Vérifiez que les fichiers existent et que vous y avez accès

Utilisez la commande **ls -ltr** pour les fichiers que vous utiliserez lors de la personnalisation. Exemple :

```
ls -ltrd /java/java71_bit64_GA/J7.1_64/bin
```

donne

```
drwxr-xr-x 4 SYSTASK TSOUSER 8192 Nov 15 2013 /java/java71_bit64_GA/J7.1_64/bin
```

où `drwxr-xr-x` signifie

d

Il s'agit d'un répertoire.

rwX

Le propriétaire *SYSTASK* dispose d'un accès en lecture, écriture et exécution au répertoire.

r-x

Les personnes du groupe *TSOUSER* peuvent lire et exécuter des fichiers dans le répertoire.

r-x

L'accès universel, c'est-à-dire que n'importe qui peut lire ou exécuter des fichiers dans le répertoire.

Vérifiez les fichiers spécifiés dans:

Chemin d'accès	Accès requis par les utilisateurs effectuant la configuration
BFG_JAVA_HOME	Lecture et exécution
/tmp	Lecture et écriture
BFG_PROD	Lecture
BFG_DATA	Ecriture
MQ_PATH	Lecture

Concepts associés

«Avant de commencer», à la page 66

La configuration Managed File Transfer (MFT) utilise des fichiers dans les fichiers UNIX System Services (USS) et PDSE.

«Configurations MFT for z/OS communes», à la page 70

Présentation des différentes configurations Managed File Transfer

Configurations MFT for z/OS communes

Présentation des différentes configurations Managed File Transfer

Managed File Transfer utilise des agents connectés à un gestionnaire de files d'attente pour le transfert de données.

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 connaître Unix System Services (USS). Exemple :
 - La structure de répertoires et les fichiers avec des noms tels que `/u/userID/myfile.txt`
 - Les commandes USS, telles que :
 - 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 USS pour pouvoir configurer et exécuter MFT :
 - Java ; par exemple, `/java/java71_bit64_GA/J7.1_64/`
 - IBM MQ V800 ; par exemple, `/mqm/V8R0M03`.
 - Bibliothèques JDBC Db2, si vous souhaitez utiliser Db2 pour le statut et l'historique ; par exemple, `/db2/db2v10/jdbc/lib`

Vous avez besoin d'un gestionnaire de files d'attente de coordination. Toutefois, vous pouvez utiliser le même gestionnaire de files d'attente pour exécuter des agents, pour traiter des commandes et pour la coordination. Si vous utilisez plusieurs gestionnaires de files d'attente, vous devez en choisir un pour agir en tant que coordinateur.

Vérifiez votre connectivité IBM MQ

Si vous disposez d'un gestionnaire de files d'attente de coordinateur MFT existant, vous avez besoin d'une connectivité entre le gestionnaire de files d'attente dans lequel vous effectuez la configuration et les gestionnaires de files d'attente de coordination et de commandes.

Configuration de MQMFTCredentials.xml sur z/OS

Si Managed File Transfer (MFT) est configuré avec la sécurité activée, l'authentification de connexion requiert tous les agents MFT, ainsi que les commandes qui se connectent à un gestionnaire de files d'attente, pour fournir les données d'identification par ID utilisateur et mot de passe.

De même, les consignateurs MFT peuvent être tenus de spécifier un ID utilisateur et un mot de passe lors de la connexion à une base de données.

Ces données d'identification peuvent être stockées dans le fichier de données d'identification MFT. Notez que les fichiers de données d'identification sont facultatifs, mais il est plus facile de définir le ou les fichiers dont vous avez besoin avant de personnaliser l'environnement.

En outre, si vous disposez de fichiers de données d'identification, vous recevez moins de messages d'avertissement. Les messages d'avertissement vous informent que MFT considère que la sécurité du gestionnaire de files d'attente est désactivée et que, par conséquent, vous ne fournissez pas de détails d'authentification.

Vous trouverez un exemple de fichier de données d'identification dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/credentials`.

Voici un exemple de fichier `MQMFTCredentials.xml`:

```
<?xml version="1.0" encoding="IBM-1047"?>
<tns:mqmftCredentials xmlns:tns="http://wmqfte.ibm.com/MFTCredentials"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://wmqfte.ibm.com/MFTCredentials MFTCredentials.xsd">
```

```

<tns:qmgr name="MQPH" user="ADMIN" mqUserId="JOHNDOEH" mqPassword="cXXXX" />
<tns:qmgr name="MQPI" user="ADMIN" mqUserId="JOHNDOEI" mqPassword="yXXXX" />
<tns:qmgr name="MQPH" mqUserId="NONEH" mqPassword="yXXXX" />
<tns:qmgr name="MQPI" mqUserId="NONEI" mqPassword="yXXXX" />
</tns:mqmftCredentials>

```

Lorsqu'un travail avec l'ID utilisateur ADMIN doit se connecter au gestionnaire de files d'attente MQPH, il transmet l'ID utilisateur *JOHNDOEH* et utilise le mot de passe *cXXXX*.

Si le travail est exécuté par un autre ID utilisateur et qu'il connecte MQPH, il transmet l'ID utilisateur *NONEH* et le mot de passe *yXXXX*.

L'emplacement par défaut du fichier *MQMFTCredentials.xml* est le répertoire de base de l'utilisateur sur z/OS Unix System Services (USS). Il est également possible de stocker le fichier dans un autre emplacement sur USS ou dans un membre d'un fichier partitionné.

Si le fichier de données d'identification est stocké à un autre emplacement, vous pouvez utiliser les propriétés suivantes pour indiquer où les commandes doivent le rechercher:

Tableau 6. : Propriétés qui définissent l'emplacement du fichier MQMFTCredentials.xml pour diverses commandes.

Type de commande	Fichier de propriétés	Nom de la propriété
Commande qui se connecte au gestionnaire de file d'attente de coordination	coordination.properties	coordinationQMGrAuthenticationCredentialsFile
Commande qui se connecte au gestionnaire de files d'attente de commandes	connection.properties	connectionQMGrAuthenticationCredentialsFile
Commande qui se connecte à un processus d'agent	agent.properties	agentQMGrAuthenticationCredentialsFile
Commande qui se connecte à un processus de consignateur	logger.properties	loggerQMGrAuthenticationCredentialsFile

Tableau 7. : Propriétés qui définissent l'emplacement du fichier MQMFTCredentials.xml pour les agents et les processus de consignateur.

Type de commande	Fichier de propriétés	Nom de la propriété
Agents MFT	agent.properties	agentQMGrAuthenticationCredentialsFile
MFT loggers	logger.properties	loggerQMGrAuthenticationCredentialsFile

Pour plus de détails sur les commandes et les processus qui se connectent à quel gestionnaire de files d'attente, voir [Quelles commandes et quels processus MFT se connectent à quel gestionnaire de files d'attente.](#)

Pour créer le fichier de données d'identification dans un fichier partitionné, procédez comme suit:

- Créez un ensemble de données partitionnées étendu avec le format VB et la longueur d'enregistrement logique (Lrecl) 200.
- Créez un membre dans le fichier, notez le fichier et le membre et ajoutez le code suivant au membre:

```

<?xml version="1.0" encoding="IBM-1047"?>
<tns:mqmftCredentials xmlns:tns="http://wmmqfte.ibm.com/MQMFTCredentials"

```

```
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://wmqfte.ibm.com/MFTCredentials MQMFTCredentials.xsd">
  <!--credentials information goes here-->
</tns:mqmftCredentials>
```

Vous pouvez protéger le fichier de données d'identification à l'aide d'un produit de sécurité, par exemple RACF, mais les ID utilisateur exécutant les commandes Managed File Transfer et administrant les processus d'agent et de consignateur ont besoin d'un accès en lecture à ce fichier.

Vous pouvez masquer les informations de ce fichier à l'aide du JCL du membre BFGCROBS. Cette opération utilise le fichier et chiffre l'ID utilisateur et le mot de passe IBM MQ . Par exemple, le membre BFGCROBS prend la ligne

```
<tns:qmgr name="MQPI" user="JOHND0E2" mqUserId="JOHND0E1" mqPassword="yXXXX" />
```

et crée

```
<tns:qmgr mqPasswordCipher="e977c61e9b9c363c" mqUserIdCipher="c394c5887867157c"
name="MQPI" user="JOHND0E2"/>
```

Si vous souhaitez conserver le mappage de l'ID utilisateur à l'ID utilisateur IBM MQ , vous pouvez ajouter des commentaires au fichier. Par exemple:

```
<!-- name="MQPI" user="ADMIN" mqUserId="JOHND0E1" -->
```

Ces commentaires ne sont pas modifiés par le processus d'obscurcissement.

Notez que le contenu est masqué et qu'il n'est pas fortement chiffré. Vous devez limiter les ID utilisateur qui ont accès au fichier.

Tâches associées

Configuration de MQMFTCredentials.xml sur Multiplatforms

Si Managed File Transfer (MFT) est configuré avec la sécurité activée, l'authentification de connexion 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. De même, les consignateurs MFT peuvent être tenus de spécifier un ID utilisateur et un mot de passe lors de la connexion à une base de données. Ces données d'identification peuvent être stockées dans le fichier de données d'identification MFT .

Copie de SBFGCMD5 pour créer une bibliothèque JCL

Vous devez créer une bibliothèque JCL pour chaque agent et consignateur. Le JCL contient la configuration et les travaux utilisés pour créer et exécuter l'agent ou le consignateur.

Pour chaque agent et consignateur, créez une copie de la bibliothèque SBFGCMD5 fournie par IBM en éditant et en exécutant le membre BFGCOPY.

Cette bibliothèque est utilisée pour définir la configuration de l'agent ou du consignateur et, après la personnalisation, contient des travaux qui peuvent être utilisés pour créer la configuration Managed File Transfer requise et l'agent ou le consignateur.

Vous créez le membre BFGCUSTM dans le cadre de ce processus.

Remarque : Si vous connaissez les commandes USS, vous pouvez configurer z/OS avec les mêmes commandes que celles que vous utilisez sur d'autres plateformes.

Concepts associés

«Configurations MFT for z/OS communes», à la page 70

Présentation des différentes configurations Managed File Transfer

«Edition du membre BFGCUSTM», à la page 74

Vous devez éditer le membre BFGCUSTM et entrer les valeurs des paramètres utilisés par votre entreprise avant d'exécuter le travail.

Edition du membre BFGCUSTM

Vous devez éditer le membre BFGCUSTM et entrer les valeurs des paramètres utilisés par votre entreprise avant d'exécuter le travail.

Voir [Paramètres requis pour le membre BFGCUSTM](#) pour obtenir la liste des paramètres nécessitant des valeurs spécifiques.

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 ne doivent pas être renseignés pour une connexion de liaisons sur la machine locale.

S'il s'agit du premier gestionnaire de files d'attente de votre environnement Managed File Transfer et que vous souhaitez utiliser le même gestionnaire de files d'attente pour la coordination, les commandes et les agents en cours d'exécution, définissez les valeurs sur le nom du gestionnaire de files d'attente local.

```
coordinationQMgr=MQPV  
connectionQMgr=MQPV
```

où MQPV est le nom de votre gestionnaire de files d'attente local.

Soumettez le travail, qui met à jour le PDSE, et crée une structure de répertoire sous le chemin indiqué.

Notez que ce travail nécessite une utilisation exclusive. Vous devez donc arrêter d'utiliser PSDE pendant l'exécution du travail.

Astuce : Chaque fois que vous soumettez le travail BFGCUSTM, le travail remplace tous les fichiers JCL. Vous devez renommer chaque membre que vous modifiez.

Concepts associés

«Avant de commencer», à la page 66

La configuration Managed File Transfer (MFT) utilise des fichiers dans les fichiers UNIX System Services (USS) et PDSE.

«Création d'un agent», à la page 76

Vous devez copier l'ensemble de données partitionnées étendu pour créer l'ensemble de données partitionnées étendu spécifique à l'agent, par exemple *user.MFT.AGENT1*. Copiez l'ensemble de données partitionnées étendu à partir d'une configuration d'agent ou de consignateur précédente, le cas échéant. S'il s'agit de votre première configuration, copiez le fichier PDSE fourni avec MFT.

Définition du gestionnaire de files d'attente de coordination

Managed File Transfer requiert la création d'un gestionnaire de files d'attente qui fait office de gestionnaire de files d'attente de coordination.

Selon la configuration que vous avez choisie, ce gestionnaire de files d'attente se trouve sur le système MVS local ou sur une autre machine. Dans le premier cas, les connexions qui lui sont associées sont des connexions de liaisons et dans le second cas, il s'agit de connexions client.

Une fois que vous avez exécuté l'étape de configuration avec succès, des membres sont configurés dans le PDSE.

Le membre BFGCFR définit le gestionnaire de files d'attente de coordination et ce travail:

1. Crée une structure de répertoires dans le répertoire Managed File Transfer (MFT) et crée des fichiers de configuration.
2. Exécute CSQUTIL pour définir les ressources IBM MQ .

Si le gestionnaire de files d'attente de coordination se trouve sur une machine distante, cette étape de travail échoue.

Le membre BCFCFR crée des fichiers dans USS et crée des définitions MQ . Ce travail:

1. Crée une rubrique MFT,
2. Crée une file d'attente MFT
3. Modifie *NAMELIST (SYSTEM.QPUBSUB.QUEUE.NAMELIST)* doit être *NAMES (SYSTEM.BROKER.DEFAULT.STREAM, SYSTEM.BROKER.ADMIN.STREAM, SYSTEM.FTE)*
4. Exécute *ALTER QMGR PSMODE (ENABLED)*

Une instruction *DISPLAY NAMELIST (SYSTEM.QPUBSUB.QUEUE.NAMELIST)* est émise avant d'effectuer la modification. Si votre NAMLIST n'est pas la valeur par défaut, vous devez modifier votre liste de noms pour ajouter SYSTEM.FTE à votre liste de noms

Renommez le membre BCFCFR avec votre propre préfixe, par exemple CCPCFR, car la repersonnalisation de ce fichier le remplace.

Editez ce membre renommé en insérant le nom de votre fichier de données d'identification. Exemple :

```
%BFGCMD CMD=fteSetupCoordination +  
-credentialsFile //'<MFTCredentialsDataSet(MemberName)>'
```

Sauvegardez et soumettez le travail. Notez que si vous devez soumettre à nouveau le travail, vous devez ajouter l'option *-f*.

Lorsque ce travail est exécuté, il répertorie les ressources IBM MQ qu'il crée. Vous devez protéger ces ressources.

```
DEFINE TOPIC('SYSTEM.FTE') TOPICSTR('SYSTEM.FTE') REPLACE  
ALTER TOPIC('SYSTEM.FTE') NPMGDLV(ALLAVAIL) PMGDLV(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)
```

Tâches associées

«Définition du gestionnaire de files d'attente de commandes», à la page 75

Vous pouvez utiliser le même gestionnaire de files d'attente que les gestionnaires de files d'attente de coordination et de commandes ou créer un nouveau gestionnaire de files d'attente de commandes.

Définition du gestionnaire de files d'attente de commandes

Vous pouvez utiliser le même gestionnaire de files d'attente que les gestionnaires de files d'attente de coordination et de commandes ou créer un nouveau gestionnaire de files d'attente de commandes.

Pourquoi et quand exécuter cette tâche

Vous devez disposer d'un gestionnaire de files d'attente de commandes, mais vous pouvez utiliser le même gestionnaire de files d'attente pour les gestionnaires de files d'attente de coordination et de

commandes. Sinon, vous devez créer un nouveau gestionnaire de files d'attente de commandes. Il peut se trouver sur la même machine que le gestionnaire de file d'attente de coordination, mais il n'est pas nécessaire qu'il le soit.

Procédure

1. Renommez le membre BFGCMCR avec votre propre préfixe, par exemple CCPCMCR.
Vous devez renommer BFGCMCR car la repersonnalisation de ce fichier le remplace.
2. Editez le membre renommé en insérant le nom de votre fichier de données d'identification.

Exemple :

```
%BFGCMD CMD=fteSetupCommands +  
-credentialsFile //'<MFTCredentialsDataSet(MemberName)>' +
```

3. Sauvegardez et soumettez le travail.
Notez que si vous devez soumettre à nouveau le travail, vous devez ajouter l'option `-f`.
Ce gestionnaire de files d'attente est utilisé pour les commandes telles que **ftePingAgent**.
4. Passez en revue ce membre, soumettez-le et passez en revue la sortie.

Que faire ensuite

Pour plus d'informations sur la création d'un agent, voir [«Création d'un agent»](#), à la page 76 .

Concepts associés

«Définition du gestionnaire de files d'attente de coordination», à la page 74

Managed File Transfer requiert la création d'un gestionnaire de files d'attente qui fait office de gestionnaire de files d'attente de coordination.

Tâches associées

«Configuration de MQMFTCredentials.xml sur plusieurs plateformes», à la page 43

Si Managed File Transfer (MFT) est configuré avec la sécurité activée, l'authentification de connexion 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. De même, les consignateurs MFT peuvent être tenus de spécifier un ID utilisateur et un mot de passe lors de la connexion à une base de données. Ces données d'identification peuvent être stockées dans le fichier de données d'identification MFT .

Référence associée

«Format de fichier de données d'identification MFT», à la page 835

Le fichier `MQMFTCredentials.xml` contient les informations d'ID utilisateur et de mot de passe sensibles. Les éléments du fichier `MQMFTCredentials.xml` doivent être conformes au schéma `MQMFTCredentials.xsd` . La sécurité des fichiers de données d'identification est de la responsabilité de l'utilisateur.

Création d'un agent

Vous devez copier l'ensemble de données partitionnées étendu pour créer l'ensemble de données partitionnées étendu spécifique à l'agent, par exemple `user.MFT.AGENT1`. Copiez l'ensemble de données partitionnées étendu à partir d'une configuration d'agent ou de consignateur précédente, le cas échéant. S'il s'agit de votre première configuration, copiez le fichier PDSE fourni avec MFT.

Réviser le membre BFGCUSTM et si vous devez utiliser un fichier de données d'identification différent, créez-en un.

Une grande partie du contenu reste identique à celui de la personnalisation détaillée dans [«Edition du membre BFGCUSTM»](#), à la page 74.

Vous devez modifier:

- `// SYSEXEC DD DSN=SCEN.FTE.JCL.AGENT1`

- LIBRARY correspondant à l'ensemble de données partitionnées étendu de l'agent
- SERVICE_TYPE=AGENT
- NAME doit être le nom de l'agent (correspondant à l'ensemble de données partitionnées étendu) JOBCARD
- Modifiez BFG_JVM_PROPERTIES = "-Xmx1024M"

Soumettez ce travail en vous rappelant que le travail requiert un accès exclusif au fichier.

Les travaux de l'agent ont tous des noms au format *BFGAG**

Renommez le membre *BFGAGCR*. Ce travail met à jour les fichiers dans le répertoire Managed File Transfer et utilise CSQUTIL pour créer des files d'attente spécifiques à l'agent dans le gestionnaire de files d'attente local. Indiquez le nom de votre fichier de données d'identification, par exemple `-credentialsFile // 'SCEN.FTE.JCL.VB(CREDOLD)`. Si vous ne spécifiez pas de nom, le travail de démarrage de l'agent n'utilise pas de fichier de données d'identification.

Vérifiez la sortie pour vous assurer que le processus s'est exécuté correctement.

Astuce : Copiez le nom de chemin du fichier *agent.properties* à partir de la sortie du travail dans un membre de l'ensemble de données partitionnées étendu pour l'agent.

Par exemple, copiez `/u/userid/fte/wmqmft/mqft/config/MQPA/agents/AGENT1/agent.properties` dans le membre AGENT.

Cela est utile si vous devez afficher le fichier de propriétés et ajouter la ligne `/u/userid/fte/wmqmft/mqft/logs/MQPA/agents/AGENT1/logs`.

Il s'agit de l'emplacement où les fichiers de trace sont stockés.

Concepts associés

«Définition du gestionnaire de files d'attente de coordination», à la page 74

Managed File Transfer requiert la création d'un gestionnaire de files d'attente qui fait office de gestionnaire de files d'attente de coordination.

«Utilisation de l'agent», à la page 77

Comment utiliser diverses commandes pour vous assurer que l'agent fonctionne correctement.

Tâches associées

«Définition du gestionnaire de files d'attente de commandes», à la page 75

Vous pouvez utiliser le même gestionnaire de files d'attente que les gestionnaires de files d'attente de coordination et de commandes ou créer un nouveau gestionnaire de files d'attente de commandes.

Utilisation de l'agent

Comment utiliser diverses commandes pour vous assurer que l'agent fonctionne correctement.

Démarrez l'agent

Renommez le membre *BFGAGST*, réviser le membre et soumettez le travail.

Si cela fonctionne, vous recevez le message *BFGAG0059I*: L'agent a été correctement démarré.

Afficher le (s) agent (s) actif (s)

Renommez le membre *BFGAGLI*, réviser le membre et soumettez le travail qui utilise le gestionnaire de files d'attente de coordination.

Vous devez résoudre les problèmes de connectivité

Envoyez une commande ping à l'agent pour vérifier qu'il fonctionne

Renommez le membre *BFGAGPI*, réviser le membre et soumettez le travail qui utilise le gestionnaire de files d'attente de commandes.

Vous devez résoudre les problèmes de connectivité

Effectuer un transfert d'essai

Pour plus d'informations, voir [«Exécution d'un transfert de vérification»](#), à la page 85.

Arrêtez l'agent

Renommez le membre BFGAGSP, révisez le membre et soumettez le travail.

Redémarrez l'agent à l'aide du membre BFGAGST.

Concepts associés

[«Création d'un agent»](#), à la page 76

Vous devez copier l'ensemble de données partitionnées étendu pour créer l'ensemble de données partitionnées étendu spécifique à l'agent, par exemple *user.MFT.AGENT1*. Copiez l'ensemble de données partitionnées étendu à partir d'une configuration d'agent ou de consignateur précédente, le cas échéant. S'il s'agit de votre première configuration, copiez le fichier PDSE fourni avec MFT.

Mise à jour d'un fichier de commandes d'agent ou de consignateur MFT existant sous z/OS

Vous pouvez mettre à jour un fichier de bibliothèque PDSE de commande Managed File Transfer créé à partir du fichier de modèle de commande Managed File Transfer .

Procédure

1. Editez le membre de script JCL BFGCUSTM et mettez à jour les variables et les propriétés dans l'instruction de définition de données BFGSTDIN.

Si vous souhaitez supprimer une propriété précédemment définie, mettez sa valeur à blanc au lieu de supprimer l'entrée. Lorsque le script JCL BFGCUSTM est exécuté, les propriétés spécifiées sont appliquées en tant que mise à jour aux fichiers de propriétés de l'agent et du consignateur UNIX System Services réels ; la définition d'une propriété sur une valeur vide indique que la propriété doit être supprimée.
2. Soumettez le travail BFGCUSTM. Ce travail génère à nouveau l'ensemble de commandes JCL correspondant à l'agent ou au consignateur. Pour la liste complète des commandes, voir [«Scripts JCL des commandes de l'agent et du consignateur z/OS»](#), à la page 83. Examinez le journal du travail de sortie pour vérifier que le script JCL s'est exécuté correctement. S'il y a des échecs, corrigez-les et soumettez à nouveau le travail BFGCUSTM.

Résultats

Vous pouvez modifier les scripts JCL générés et ajouter votre propre logique. Cependant, soyez prudent lorsque vous exécutez à nouveau BFGCUSTM car vous risquez d'écraser la logique personnalisée.

Concepts associés

[«Options de configuration MFT sous z/OS»](#), à la page 60

Les options de configuration Managed File Transfer sous z/OS sont identiques à celles des plateformes réparties.

Tâches associées

[«Création d'un fichier de commandes d'agent ou de consignateur MFT»](#), à la page 64

Vous pouvez créer un jeu de données PDSE de commandes à partir du jeu de données du modèle de commande Managed File Transfer pour un Managed File Transfer Agent ou Managed File Transfer Logger spécifique pour une coordination spécifique.

z/OS variables JCL

Vous pouvez utiliser des valeurs de substitution, des variables JCL et des propriétés de configuration dans le script BFGCUSTM.

Le tableau suivant répertorie les valeurs de substitution pour le script JCL BFGCUSTM dans un fichier de bibliothèque PDSE de commande MFT . Vous devez remplacer ces valeurs de substitution par des valeurs appropriées avant de soumettre le travail BFGCUSTM.

<i>Tableau 8. Valeurs de substitution pour le script JCL BFGCUSTM</i>	
Variable de substitution	Valeur
++library++	Nom de fichier de la bibliothèque PDSE de la commande MFT qui le contient.
+ + bfg_java_home + +	Emplacement de votre installation Java .
+ + bfg_prod + +	Emplacement du répertoire racine des services système UNIX d'installation du produit MFT .

Le tableau suivant décrit les variables d'environnement de l'instruction de définition de données BFGSTDIN pour le script JCL BFGCUSTM, dans un fichier de bibliothèque PDSE de commande MFT (dans la section [Variables]). Vous devez remplacer toutes les variables spécifiées par des valeurs de substitution (c'est-à-dire, les valeurs placées entre deux signes plus, + +) par des valeurs appropriées avant de soumettre le travail BFGCUSTM.

<i>Tableau 9. Variables d'environnement</i>	
Variable d'environnement	Valeur
LIBRARY	Nom de fichier de la bibliothèque PDSE de la commande MFT qui le contient.
TMPDIR	Répertoire des services système UNIX pour les fichiers temporaires.
BFG_PROD	Emplacement du répertoire racine des services système UNIX d'installation du produit MFT .
BFG_DATA	Emplacement du répertoire de données pour Managed File Transfer for z/OS, qui correspond au chemin d'accès à <i>DATA_DIR</i> .
BFG_JAVA_HOME	Emplacement de votre installation Java .
BFG_JVM_PROPERTIES	Facultatif. Définit une valeur pour la variable d'environnement BFG_JVM_PROPERTIES. Ces propriétés sont transmises à la machine virtuelle Java .

Tableau 9. Variables d'environnement (suite)

Variable d'environnement	Valeur
BFG_GROUP_NAME	<p>Le groupe de fichiers mqm est généralement associé aux fichiers de données de configuration et aux commandes MFT . Par conséquent, tous les utilisateurs membres du groupe mqm peuvent accéder à la configuration MFT et y apporter des modifications. Pour plus d'informations, voir la section «Droits d'accès au système de fichiers pour MFT dans IBM MQ», à la page 359.</p> <p>Pour un système z/OS , un groupe de fichiers est une entité de système de fichiers USS et le groupe de fichiers mqm n'est pas nécessairement défini. Vous pouvez associer un groupe de systèmes de fichiers z/OS USS pour les fichiers de données de configuration MFT à l'aide de la variable d'environnement BFG_GROUP_NAME. Par exemple, à l'invite de l'interpréteur de commandes USS, utilisez:</p> <pre data-bbox="862 808 1472 884">export BFG_GROUP_NAME=FTEGB</pre> <p>qui définit un groupe <i>FTEGB</i> à associer à tous les fichiers de configuration créés ultérieurement pour la session USS en cours.</p> <p>Vous pouvez définir BFG_GROUP_NAME sur une valeur vide ou la supprimer.</p> <p>Remarque : Lors de la première exécution de BFGCUSTOM, si la configuration MFT doit être utilisée par plusieurs ID utilisateur, il est important que BFG_GROUP_NAME soit défini sur un groupe accessible à tous les ID utilisateur requis. Si BFGCUSTOM est réexécuté, BFG_GROUP_NAME ne doit pas être modifié (sinon, les droits d'accès au fichier de groupe USS pour tous les fichiers et répertoires du répertoire référencé par BFG_DATA doivent également être modifiés pour refléter le nouveau paramètre BFG_GROUP_NAME).</p> <p>V9.0.0.1 Si vous exécutez la commande fteMigrateAgent sur un système z/OS avec la variable d'environnement BFG_GROUP_NAME définie sur une valeur non vide, la commande vérifie si l'utilisateur est membre du groupe nommé par la variable BFG_GROUP_NAME. Si l'utilisateur ne fait pas partie du groupe nommé, la commande peut signaler le message d'erreur BFGCL0502E: Vous n'êtes pas autorisé à effectuer l'opération demandée. et ne parvient pas à s'exécuter. Pour plus de détails sur les critères que l'utilisateur doit remplir pour exécuter correctement cette commande, voir «Agent fteMigrate: migration d'un agent FTE 7.0 vers MQ 7.5 ou version ultérieure», à la page 494.</p>

Tableau 9. Variables d'environnement (suite)

Variable d'environnement	Valeur
BFG_WTO	La journalisation z/OS est activée lorsque BFG_WTO est défini sur YES, ON ou TRUE. Cette commande contrôle si les messages qui sont écrits dans le journal des événements de l'agent sont également écrits dans la fonction de journal de l'opérateur z/OS , qui permet un accès plus facile aux produits d'automatisation lorsque vous exécutez un agent à partir de JCL. Le code de routage est Programmer Information (11) et le code descripteur est Informational (12).
SERVICE_TYPE	Indique si la bibliothèque de commandes MFT est destinée à un agent ou à un consignateur. Les valeurs valides sont AGENT ou LOGGER.
NOM	Nom de l'agent ou du consignateur pour la valeur SERVICE_TYPE.
QMGR	Nom du gestionnaire de files d'attente local associé à l'agent ou au consignateur pour la valeur SERVICE_TYPE.
CLASSE DE SORTIE	Classe de sortie des fichiers SYSOUT. La valeur par défaut est *, qui demande la même classe de sortie que le paramètre MSGCLASS dans l'instruction de travail.
MQ_PATH	Utilisé dans BFGPROF pour créer la variable d'environnement LIBPATH.
MQ_HLQ	Qualificatif de haut niveau pour des ensembles de données IBM MQ.
MQ_LANG	Langue requise.
DB2_HLQ	Facultatif. Qualificatif de haut niveau pour les fichiers Db2 .
JOB CARD1	Ligne d'en-tête 1 pour un travail de commande JCL.
JOB CARD2	Ligne d'en-tête 2 pour un travail de commande JCL.
JOB CARD3	Ligne d'en-tête 3 pour un travail de commande JCL.
ADMIN_JOB1	Ligne d'en-tête 1 pour un travail d'administration.
ADMIN_JOB2	Ligne d'en-tête 2 pour un travail d'administration.
ADMIN_JOB3	Ligne d'en-tête 3 pour un travail d'administration.
FTE_CONFIG	Configuration WMQFTE existante pour la migration. Indiquez une valeur vide si la migration n'est pas requise.

Variable d'environnement	Valeur
CHEMIN_DONNÉES_D'ACCÈS	Chemin d'accès au fichier de données d'identification pour la migration, par exemple /u/user1/agent3. Les fichiers de données d'identification pour la migration pour Managed File Transfer doivent se trouver dans un fichier distinct pour les informations de configuration et les fichiers de configuration sur IBM WebSphere MQ File Transfer Edition 7.0.4.4. Requis pour les commandes de migration BFGAGMG et les scripts JCL BFGLGMG uniquement. Indiquez une valeur vide si la migration n'est pas requise.
DB_PROPS_PATH	Indique le fichier de propriétés du consignateur de base de données pour la migration. Cette option est requise uniquement si le fichier de propriétés n'utilise pas le nom et le chemin d'accès par défaut suivants: config_directory/coordination_qmgr/databaselogger.properties. Indiquez une valeur vide si la migration n'est pas requise.

Remarque : Les fichiers JAR IBM MQ fournis avec MFT, dans le répertoire *MQMFT product root/java/lib*, sont toujours utilisés et ne sont pas configurables.

Le tableau suivant décrit les propriétés de configuration MFT obligatoires pour l'instruction de définition de données BFGSTDIN pour le script JCL BFGCUSTM dans un fichier de bibliothèque PDSE de commande MFT . Vous devez remplacer les propriétés spécifiées par des valeurs de substitution (c'est-à-dire les valeurs placées entre deux signes plus, + +) par une valeur non vide appropriée avant de soumettre le travail BFGCUSTM. Ces propriétés définissent des substitutions pour les propriétés de configuration MFT . Vous pouvez ajouter des propriétés d'agent et de consignateur pour personnaliser des agents ou des consignateurs pour votre environnement. Pour obtenir la liste de toutes les propriétés, voir «Fichiers de propriétés de configuration», à la page 91.

Propriété	Valeur
coordinationQMgr	Nom du gestionnaire de files d'attente de coordination pour la configuration à laquelle l'agent ou le consignateur est associé.
Hôte coordinationQMgr	Facultatif. Nom d'hôte du système sur lequel s'exécute le gestionnaire de files d'attente de coordination. Si vous laissez la valeur de cette propriété vide, une connexion en mode liaisons est utilisée.
Port coordinationQMgr	Facultatif. Numéro de port sur lequel le gestionnaire de files d'attente de coordination est en mode écoute. Ce paramètre est utilisé uniquement si vous spécifiez également une valeur non vide pour la propriété d'hôte coordinationQMgr.

Tableau 10. Propriétés de configuration obligatoires pour l'instruction de définition de données BFGSTDIN (suite)

Propriété	Valeur
Canal coordinationQMgr	Facultatif. Canal à utiliser pour la connexion au gestionnaire de files d'attente de coordination. Ce paramètre est utilisé uniquement si vous spécifiez également une valeur non vide pour la propriété d'hôte coordinationQMgr.
connectionQMgr	Nom du gestionnaire de files d'attente de commandes pour la configuration à laquelle l'agent ou le consignateur est associé.
Hôte connectionQMgr	Facultatif. Nom d'hôte du système sur lequel s'exécute le gestionnaire de files d'attente de commandes. Si vous laissez la valeur de cette propriété vide, une connexion en mode liaisons est utilisée.
connectionQMgrPort	Facultatif. Numéro de port sur lequel le gestionnaire de files d'attente de commandes est en mode écoute. Ce paramètre est utilisé uniquement si vous spécifiez également une valeur non vide pour la propriété d'hôte connectionQMgr.
Canal connectionQMgr	Facultatif. Canal à utiliser pour la connexion au gestionnaire de files d'attente de commandes. Ce paramètre est utilisé uniquement si vous spécifiez également une valeur non vide pour la propriété d'hôte connectionQMgr.

z/OS Scripts JCL des commandes de l'agent et du consignateur z/OS

Ensemble de commandes JCL disponibles dans un fichier de bibliothèque PDSE de commandes MFT .

Tableau 11. Commandes JCL disponibles dans un fichier de bibliothèque PDSE de commandes MFT

Membre	Description ou commande de ligne de commande fte
BFGCOPY	Travail de création d'une copie de cette bibliothèque
BFGCUSTM	Travail de personnalisation de cette bibliothèque pour l'agent ou le consignateur
BFGCFR	fteSetupCoordination
BFGCMCR	fteSetupCommands
BFGAGCR	AgentfteCreate . Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur AGENT.
BFGLGCRS	fteCreateLogger . Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur LOGGER.
BFGAGST	AgentfteStart . Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur AGENT.

Tableau 11. Commandes JCL disponibles dans un fichier de bibliothèque PDSE de commandes MFT (suite)

Membre	Description ou commande de ligne de commande fte
BFGAGSTP	fteStartAgent procédure. Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur AGENT.
BFGAGPI	AgentftePing. Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur AGENT.
BFGAGSP	fteStopAgent. Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur AGENT.
BFGLGST	fteStartLogger. Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur LOGGER.
BFGLGSTP	fteStartLogger procédure. Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur LOGGER.
BFGLPGS	fteStopLogger. Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur LOGGER.
BFGAGSH	fteShowAgentDetails. Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur AGENT.
BFGLGSH	fteShowLoggerDetails. Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur LOGGER.
BFGCFDF	fteChangeDefaultConfigurationOptions
BFGAGCL	AgentfteClean. Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur AGENT.
BFGAGDE	AgentfteDelete. Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur AGENT.
BFGLGDE	fteDeleteLogger. Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur LOGGER.
BFGPRSH (hôte)	fteDisplayVersion
BFGAGLI	fteListAgents. Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur AGENT.
BFGMCLI	fteListMonitors
BFGSTLI	fteListScheduledTransfers
BFGTMLI	fteListTemplates
BFGAGMG	fteMigrateAgent : Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur AGENT.
BFGLGMG	fteMigrateLogger : Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur LOGGER.
BFGCROBS	fteObfuscate exemple

Tableau 11. Commandes JCL disponibles dans un fichier de bibliothèque PDSE de commandes MFT (suite)

Membre	Description ou commande de ligne de commande fte
BFGRAS	fteRAS
BFGAGTC	fteSetAgentTraceNiveau. Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur AGENT.
BFGLGTC	«fteSetLoggerTraceLevel», à la page 520: Créé uniquement lorsque vous définissez la variable SERVICE_TYPE sur LOGGER.
BFGPRANS	fteAnt exemple
BFGTRCAS	fteCancelTransfer exemple
BFGMNCRS	fteCreateMonitor exemple
BFGTMCRS	fteCreateTemplate exemple
BFGTRCRS	fteCreateTransfer exemple
BFGMNDES	fteDeleteMonitor exemple
BFGSTDES	fteDeleteScheduledTransfer exemple
BFGTMDES	fteDeleteTemplates exemple

z/OS

Remarque : Le JCL, pour les commandes qui créent des scripts MQSC ou font référence à des scripts de suppression, vous demande d'exécuter un script, mais le script a déjà été exécuté par le travail.

z/OS

Exécution d'un transfert de vérification

Comment effectuer un transfert pour vérifier que le produit fonctionne correctement.

Renommez et éditez le membre BFGTRCRS.

1. Ajoutez un /* avant le %BFGCMD CMD=fteCreateTransfer -h
2. Supprimez les autres commentaires du membre.
3. Indiquez le nom de l'agent en cours pour -sa et -da
4. Sauvegardez le JCL
5. Soumission du JCL

Ce JCL se connecte au gestionnaire de files d'attente de commandes.

z/OS

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. Pour plus d'informations, voir «[Configuration de MQMFTCredentials.xml sur z/OS](#)», à la page 71.

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 87

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 +
-loggerQMGR 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 `BFGLCRS`.

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:

Tableau 12. Db2Variables	
Nom Db2	Exemple
<code>-dbName databaseName</code>	Vous pouvez l'obtenir à partir de la valeur d'emplacement dans le message <code>DSNL004I</code> pour votre sous-système Db2
<code>-dbDriver filePath</code>	Par exemple <code>/db2/db2v10/jdbc/classes/db2jcc.jar</code>
<code>-dbLib filePath</code>	Par exemple <code>/db2/db2v10/jdbc/lib/libdb2jcc2zos_64.so</code>

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 USS mqft/sql/ftelog_tables_zos.sql.

Créez un membre Db2 dans votre PDSE. Editez ce membre et utilisez la commande COPY sur la ligne de commande. Effectuez une copie à partir du fichier de définitions USS.

Etant 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 dont la taille de page est 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.

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.

Variables d'environnement pour MFT sous z/OS

Si vous exécutez des commandes directement à partir de l'environnement USS ou de vos propres scripts JCL, après la personnalisation et la configuration, vous devez définir un certain nombre de variables d'environnement avant d'exécuter les scripts de configuration et d'administration fournis par Managed

File Transfer. Vous devez définir ces variables pour chaque utilisateur et dans chaque environnement à partir duquel les scripts seront appelés.

Pour éviter les conflits avec d'autres produits, vous pouvez choisir de créer un script `.wmqfterc` dans votre répertoire de base. Le script `.wmqfterc` est ensuite appelé par chacun des scripts Managed File Transfer et vous pouvez l'utiliser pour fournir des paramètres d'environnement personnalisés pour Managed File Transfer.

Il existe également une variable d'environnement facultative, `BFG_WTO`, que vous pouvez définir pour envoyer des messages au journal de l'opérateur lors de l'exécution d'agents à partir de JCL.

Tableau 13. Variables d'environnement z/OS requises	
Variable d'environnement	Valeur
BFG_JAVA_HOME	Emplacement de votre installation Java . Pour plus d'informations sur les niveaux de Java pris en charge, voir Configuration système requise pour IBM MQ .
BFG_DATA	Emplacement du répertoire de données pour Managed File Transfer for z/OS. Il s'agit du chemin d'accès à <code>DATA_DIR</code> .
STEPLIB	Doit inclure les fichiers IBM MQ suivants: <ul style="list-style-type: none"> • SCSQAUTH • SCSQANLE • SCSQLOAD Si vous souhaitez exécuter le composant consignateur de base de données sur un système z/OS , STEPLIB doit également inclure les fichiers Db2 suivants dans l'ordre indiqué: <ul style="list-style-type: none"> • SDSNEXIT • SDSNLOD2 • CHARGE DSN
LIBPATH	Doit inclure l'emplacement de vos bibliothèques IBM MQJava dans l'espace z/OS UNIX System Services (pour IBM MQ 8.0, la valeur par défaut est <code>/mqm/V8R0M0/java/lib</code>).

Voici un exemple de fichier `.profile` qui configure correctement les variables d'environnement pour Managed File Transfer:

```
LIBPATH=/mqm/V8R0M0/java/lib:$LIBPATH
STEPLIB=MQM.V800.SCSQAUTH:MQM.V800.SCSQANLE:MQM.V800.SCSQLOAD
PATH=/u/fteuser/bin:/u/fteuser/J7.0/bin:/bin:/usr/bin:/u/fteuser/extras/bin:/bin:$PATH
BFG_JAVA_HOME=/u/fteuser/J7.0
BFG_DATA=/u/fteuser/DATA_DIR
export PATH LIBPATH STEPLIB BFG_JAVA_HOME BFG_DATA
```

Si vous le souhaitez, vous pouvez également définir les variables d'environnement suivantes:

Tableau 14. Variable d'environnement z/OS facultative

Variable d'environnement	Valeur
BFG_WTO	<p>L'une des valeurs suivantes active BFG_WTO:</p> <ul style="list-style-type: none">• YES• Oui• TRUE <p>L'une des valeurs suivantes désactive BFG_WTO. Ces valeurs ne sont pas sensibles à la casse.</p> <ul style="list-style-type: none">• NULL• NO• Non• FALSE <p>Active la journalisation z/OS . Par défaut, cette variable d'environnement est désactivée.</p> <p>Les messages consignés dans le journal des événements de l'agent sont également consignés dans la fonction de journal de l'opérateur z/OS , qui facilite l'accès des produits d'automatisation lorsque vous exécutez un agent à partir de JCL. Le code de routage est Programmer Information (11) et le code descripteur est Informational (12).</p>

Tableau 14. Variable d'environnement z/OS facultative (suite)

Variable d'environnement	Valeur
BFG_GROUP_NAME	<p>Le groupe de fichiers mqm est généralement associé aux fichiers de données de configuration et aux commandes Managed File Transfer . Par conséquent, tous les utilisateurs qui sont membres du groupe mqm peuvent accéder à la configuration Managed File Transfer et y apporter des modifications. Pour plus d'informations, voir «Droits d'accès au système de fichiers pour MFT dans IBM MQ», à la page 359.</p> <p>Pour un système z/OS , un groupe de fichiers est une entité de système de fichiers USS et le groupe de fichiers mqm n'est pas nécessairement défini. Vous pouvez définir un autre groupe de systèmes de fichiers z/OS USS existant pour les fichiers de données de configuration Managed File Transfer à l'aide de la variable d'environnement BFG_GROUP_NAME. Par exemple, à l'invite de l'interpréteur de commandes USS:</p> <pre data-bbox="862 846 1472 919">export BFG_GROUP_NAME=FTEGB</pre> <p>qui définit le groupe FTEGB à associer à tous les fichiers de configuration créés ultérieurement pour la session USS en cours.</p> <p>Vous pouvez définir BFG_GROUP_NAME sur une valeur vide ou la supprimer.</p> <p>V9.0.0.1 Si vous exécutez la commande fteMigrateAgent sur un système z/OS avec la variable d'environnement BFG_GROUP_NAME définie sur une valeur non vide, la commande vérifie si l'utilisateur est membre du groupe nommé par la variable BFG_GROUP_NAME. Si l'utilisateur ne fait pas partie du groupe nommé, la commande peut signaler le message d'erreur BFGCL0502E: Vous n'êtes pas autorisé à effectuer l'opération demandée. et ne parvient pas à s'exécuter. Pour plus de détails sur les critères que l'utilisateur doit remplir pour exécuter correctement cette commande, voir «Agent fteMigrate: migration d'un agent FTE 7.0 vers MQ 7.5 ou version ultérieure», à la page 494.</p>

Fichiers de propriétés de configuration

Récapitulatif des propriétés utilisées dans Managed File Transfer.

- «Le fichier MFT coordination.properties», à la page 553
- «Le fichier MFT command.properties», à la page 559
- «Le fichier MFT agent.properties», à la page 564

- [Fichier de propriétés de configuration du consignateur](#)

Configuration de MFT pour z/OS Automatic Restart Manager (ARM)

Managed File Transfer est une application compatible ARM.

Avant de commencer

Pour plus d'informations sur l'activation d'ARM et la définition de règles ARM pour votre système, voir [Utilisation d' z/OS Automatic Restart Manager \(ARM\)](#).

Si vous souhaitez utiliser la fonction MFT DB Logger pour redémarrer automatiquement et vous reconnecter à une base de données Db2 , ARM est le seul gestionnaire de redémarrage pris en charge disponible.

Pourquoi et quand exécuter cette tâche

A l'aide d'ARM, les agents et les consignateurs peuvent être configurés pour le redémarrage en définissant les propriétés d'agent/consignateur armELEMTYPEet armELEMENT. La propriété armELEMTYPE définit le type d'élément ARM et la propriété armELEMENT est le nom de l'élément que ARM doit enregistrer:

- Vous pouvez définir l'agent ELEMTYPE sur SYSBFGAG et armELEMENT peut être défini pour correspondre au nom de l'agent.
- Vous pouvez définir le paramètre ELEMTYPE du consignateur sur SYSBFGLG et armELEMENT peut être défini pour correspondre au nom du consignateur.

Remarque : Les agents et les consignateurs configurés pour être redémarrés par ARM ne peuvent être exécutés qu'à partir d'un travail par lots ou d'une tâche démarrée. Les tentatives de démarrage direct de l'agent ou du consignateur à partir de la ligne de commande USS échoueront avec un code raison d'erreur ARM.

Exemple

L'exemple suivant de règle de redémarrage définit l'agent BFGFT7CAG1 comme dépendant du gestionnaire de files d'attente FT7C:

```
RESTART_ORDER
  LEVEL (3)
  ELEMENT_TYPE (SYSBFGAG, SYSBFGLG)

RESTART_GROUP (GROUP7C)
  ELEMENT (SYSMQMGRFT7C)
  ELEMENT (BFGFT7CAG1)
  RESTART_ATTEMPTS (3, 300)
```

Utilisation de Managed File Transfer for z/OS avec le lanceur JZOS Java

Vous pouvez appliquer les instructions de cette rubrique comme méthode alternative d'utilisation de Managed File Transfer dans votre entreprise, sur votre système IBM MQ for z/OS .

Présentation

Managed File Transfer for z/OS (MFT) utilise la procédure d'installation standard de z/OS . Une autre méthode d'exécution des commandes MFT consiste à utiliser JCL et le programme de lancement Java JZOS.

Voir [JZOS Batch Launcher and Toolkit](#) pour plus de détails.

Si votre JCL ne parvient pas à s'exécuter correctement, voir [«Problèmes MFT courants avec JZOS»](#), à la page 308.

Exemple de JCL pour IBM MQ 8.0 et versions ultérieures



Avertissement : Pour IBM WebSphere MQ File Transfer Edition 7.0, les paramètres commencent par FTE_ au lieu de BFG_.

```
//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
-agentQMqr MQPD
-p MQPD
/*
```

où :

- (1) est l'emplacement des instructions JCL incluses
- (2) Inclure le membre JCL spécifié à partir de l'emplacement dans 1)
- (2A) Cela étend // STDENV-voir ci-dessous
- (3) Il s'agit de la commande à exécuter, sans le préfixe fte de début
- (4) Ces lignes sont nécessaires, ils mettent en place des informations pour JZOS
- (5) Les paramètres de la commande
- Le membre BFGJCL8 (vous pouvez sélectionner votre propre nom) appelle JZOS. Ce membre possède la bibliothèque STEPLIB et d'autres JCL nécessaires à l'exécution de MFT.

Autres JCL que vous devez inclure

Vous devez inclure le JCL pour les bibliothèques IBM MQ for z/OS et, si vous utilisez le consignateur Db2 , les bibliothèques Db2 .

Exemple :

```
//WMQFTE EXEC PGM=JVMLDM86,REGION=0M PARM='+T' (1)
//STEPLIB DD DSN=SYS1.SIEALNKE,DISP=SHR (2)
//* MQ libraries
// DD DSN=MQM.V800.SCSQAUTH,DISP=SHR MQ Bindings
// DD DSN=MQM.V800.SCSQANLE,DISP=SHR MQ Bindings
// DD DSN=MQM.V800.SCSQLOAD,DISP=SHR MQ Bindings

//* DB2 libraries
// DD DISP=SHR,DSN=SYS2.DB2.V10.SDSNEXIT.DBCP
// DD DISP=SHR,DSN=SYS2.DB2.V10.SDSNLOAD
// DD DISP=SHR,DSN=SYS2.DB2.V10.SDSNLOD2
//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)
```

où :

- (1) est le nom du programme JZOS. Recherchez la version de votre système dans SYS1.SIEALNKE . Ajoutez, PARM =+ T'pour fournir des diagnostics supplémentaires.
- (2) Il s'agit de l'ensemble de données avec le programme JZOS.
- (3) Il s'agit du nom de membre d'un script shell. Il définit les paramètres requis par MFT. Voir [«Script shell permettant de définir MFT»](#), à la page 94.

Il peut s'agir de n'importe quel fichier et membre. Il doit être le dernier dans le fichier car le travail JCL l'étend. Voir 2A dans la [«Exemple de JCL pour IBM MQ 8.0 et versions ultérieures»](#), à la page 93.

Script shell permettant de définir MFT

Dans l'exemple «Autres JCL que vous devez inclure», à la page 93 , le membre BFGZENV8 est utilisé. Il est basé sur le profil JZOS.

Vous pouvez utiliser le même fichier de configuration pour MFT V8 et IBM WebSphere MQ File Transfer Edition 7.0, avec quelques modifications mineures. N'oubliez pas qu'avant MFT V8 , les paramètres commencent par FTE. Voir «Exemple de fichier», à la page 94.

Vous devez savoir:

- Emplacement dans lequel Java est installé
- Emplacement des bibliothèques IBM MQ for z/OS Java
- Emplacement des fichiers MFT
- Un ID utilisateur doit faire partie d'un groupe spécifique pour être considéré comme un administrateur IBM MQ for z/OS . Vous avez besoin du nom de ce groupe
- Si vous n'utilisez pas l'anglais pour les messages, vous devez connaître la langue à spécifier.

Exemple de fichier

```
# 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/V8R0M0/java/lib/"
export LIBPATH

export BFG_JAVA_HOME="${JAVA_HOME}"
export BFG_WTO="YES"
export BFG_GROUP_NAME=MQADM
export BFG_PROD="/HMF8800/"
export BFG_CONFIG="/u/johndoe/fteconfig"
# export BFG_LANG=" -Duser.language=de "
export BFG_LANG=" "
```

où :

export _BPXK_AUTOCVT = "ON "

Est requis pour la conversion Unicode

export JAVA_HOME = "/java/java71_bit64/J7.1_64/"

Correspond à l'emplacement du répertoire Java. Indiquez le nom du chemin d'accès à Java. Ce répertoire contient bin et d'autres répertoires.

export PATH= "/bin: \${JAVA_HOME}/bin/classic/"

Configure l'instruction de chemin pour les instructions exécutables Java

LIBPATH= "/lib:/usr/lib:\${JAVA_HOME}/bin"

Configure le chemin d'accès à la bibliothèque pour les instructions exécutables Java

LIBPATH= " \$LIBPATH: \${JAVA_HOME}/bin/classic"

Ajoute d'autres bibliothèques Java à l'instruction LIBPATH.

LIBPATH=\$LIBPATH: "/mqm/V8R0M0/java/lib/"

Ajoute des bibliothèques IBM MQ for z/OS dans le chemin d'accès à la bibliothèque. Indiquez le nom de vos bibliothèques IBM MQ for z/OS dans USS.

export LIBPATH

Rend le LIBPATH disponible pour JZOS

export BFG_JAVA_HOME = "\${JAVA_HOME}"

Définit BFG_JAVA_HOME sur la valeur de JAVA_HOME spécifiée ci-dessus

export BFG_WTO = "YES "

La définition de BFG_WTO sur YES entraîne l'affichage de messages dans le journal des travaux à l'aide de WTO.

export BFG_GROUP_NAME=MQADM

Les ID utilisateur, qui sont membres du groupe spécifié, sont considérés comme des administrateurs IBM MQ for z/OS

export BFG_PROD = "/HMF8800/"

Est le chemin dans lequel se trouve le code MFT

export BFG_DATA = "/u/johndoe/fteconfig"

Emplacement de stockage des informations de configuration MFT

export BFG_LANG = " -Duser.language= de"

Est une instruction mise en commentaire pour définir la langue comme étant l'allemand

export BFG_LANG = ""

Indique la langue par défaut, l'anglais.

Le contenu du produit MFT dans `/lib/messages/BFGNVMessages_*.properties` répertorie les langues disponibles. La valeur par défaut est de laisser la valeur vide, ce qui signifie que l'anglais est utilisé.

Pour V7 , indiquez:

```
export FTE_JAVA_HOME="${JAVA_HOME}"
export FTE_WTO="YES"
export FTE_GROUP_NAME=SCENU
export FTE_PROD="/HMF7100/"
export FTE_CONFIG="/u/johndoe/fteconfig"
export BFG_LANG=""
```

Notez que `/u/johndoe/fteconfig` est différent de celui de `BFG_DATA`

Information associée

[Managed File Transfer for z/OS](#)

[Planification de 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 la section «[Configuration du gestionnaire de files d'attente de coordination pour MFT](#)», à la page 99.

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 la section «[Configuration des gestionnaires de files d'attente d'agent MFT](#)», à la page 100.

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 SBSDB(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](#)», à la page 46.

Concepts associés

«[Configuration de MFT pour la première utilisation](#)», à la page 97

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.

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, **ftCreateTransfer**) 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 104](#)

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 98

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 99

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](#).

- 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. Si la communication avec Managed File Transfer dans les agents IBM WebSphere MQ 7.5 ou Managed File Transfer est requise, vérifiez que le paramètre CONVERT du canal émetteur est défini sur no. (Les versions antérieures de IBM WebSphere MQ File Transfer Edition publiaient toujours les messages au format UTF-8 , ce qui signifie que toute conversion de données endommage le message. Cela n'est pas nécessaire pour les agents sous Managed File Transfer dans IBM MQ 8.0 ou version ultérieure, car 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

Ensuite, suivez les étapes de configuration pour votre gestionnaire de file d'attente de coordination : [Configuration du gestionnaire de file 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 (NPMGDLV) 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 97

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 104

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 98

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», à la page 527

La commande **fteSetupCoordination** crée des fichiers de propriétés ainsi que le répertoire du gestionnaire de file d'attente de coordination pour Managed File Transfer.

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, voir [«Paramètres de file d'attente de l'agent MFT»](#), à la page 637.

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, voir [«Paramètres de file d'attente de l'agent MFT»](#), à la page 637.

Que faire ensuite

Pour plus d'informations sur la création et la configuration d'un agent de pont de protocole, voir [«fteCreateBridgeAgent \(créer et configurer un agent de pont de protocole MFT\)»](#), à la page 403 et [«Configuration d'un pont de protocole pour un serveur FTPS»](#), à la page 235.

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 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 de connexion MFT et IBM MQ»](#), à la page 42.

Remarques :

- Pour localiser votre répertoire de configuration MFT , utilisez la commande **fteDisplayVersion -v** .
- Pour les utilisateurs z/OS , le fichier MQMFTCcredential.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:
-credentialsFile *full credential file path*.
- Le mot de passe en texte clair dans MQMFTCcredential.xml peut être brouillé à l'aide de la commande suivante:

```
fteObfuscate -credentialsFile full file path to MQMFTCcredentials.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.

Référence associée

«fteSetupCoordination», à la page 527

La commande **fteSetupCoordination** crée des fichiers de propriétés ainsi que le répertoire du gestionnaire de file d'attente de coordination pour Managed File Transfer.

«Commandes fteSetup: créez le fichier MFT *command.properties*», à la page 524

La commande **fteSetupCommands** crée le fichier Managed File Transfer *command.properties*. Ce fichier de propriétés spécifie les détails du gestionnaire de files d'attente qui se connecte au réseau IBM MQ lorsque vous émettez des commandes.

«Agent fteCreate(créer un agent MFT)», à la page 399

La commande **fteCreateAgent** crée un Managed File Transfer Agent et sa configuration associée.

«fteObfuscate: chiffrement des données sensibles», à la page 507

La commande **fteObfuscate** chiffre les données sensibles dans les fichiers de données d'identification. Ainsi, le contenu des fichiers de données d'identification n'est plus lu par une personne qui a accès au fichier.

«Format de fichier de données d'identification MFT», à la page 835

Le fichier `MQMFTcredentials.xml` contient les informations d'ID utilisateur et de mot de passe sensibles. Les éléments du fichier `MQMFTcredentials.xml` doivent être conformes au schéma `MQMFTcredentials.xsd`. La sécurité des fichiers de données d'identification est de la responsabilité de l'utilisateur.

«Le fichier MFT `agent.properties`», à la page 564

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.

«Que faire si votre agent MFT n'est pas répertorié par la commande `fteListAgents ?`», à la page 309

Si votre agent n'est pas répertorié par la commande **fteListAgents** ou n'est pas affiché dans IBM MQ Explorer, ou si vos transferts de fichier ne sont pas affichés dans le **journal de transfert** de IBM MQ Explorer, vous pouvez effectuer un certain nombre d'étapes afin d'identifier le problème.

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 [Multi-instance queue managers](#). Veuillez 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`», à la page 564.

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»](#), à la page 553.

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 129. 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 [Multi-instance queue managers](#).

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 .

Pour plus d'informations, voir [«Le fichier MFT command.properties»](#), à la page 559.

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 97

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 98

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 99

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.

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 consigneur de base de données, voir [«Configuration d'un consigneur MFT»](#), à la page 107.

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 avec 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, vous pouvez spécifier que le plug-in d'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.

Ecriture 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 d'un consigneur 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 consigneur 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:

- consignateur de fichiers autonome
- consignateur de base de données autonome
- Consignateur Java Platform, Enterprise Edition (Java EE)

Important : 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** .

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 des instructions à ce sujet, voir [«Installation du consignateur de fichiers MFT autonome»](#), à la page 109.

Le consignateur de fichiers autonome n'est pas pris en charge sous z/OS ou 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.

Si vous utilisez un système Windows , vous pouvez exécuter les consignateurs autonomes en tant que services Windows pour vous assurer que les consignateurs continuent de s'exécuter lorsque vous vous déconnectez de votre session Windows . Pour obtenir des instructions, voir [«Installation du consignateur de base de données autonome MFT»](#), à la page 116 pour un consignateur de base de données autonome.

Java EE CONSIGNATEUR DE BASE DE DONNEES

Le consignateur 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 consignateur de base de données autonome si vous disposez d'un environnement de serveur d'applications Java EE existant car le consignateur de base de données Java EE peut être géré en même temps que vos autres applications d'entreprise. Vous pouvez également installer le consignateur de base de données Java EE sur un système distinct sur les systèmes hébergeant votre serveur et votre base de données IBM MQ . Le consignateur de base de données Java EE est pris en charge pour une utilisation avec les bases de données Db2 et Oracle . Le consignateur 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 consignateur, voir les rubriques suivantes:

- [«Installation du consignateur de fichiers MFT autonome»](#), à la page 109
- [«Installation du consignateur de base de données autonome MFT»](#), à la page 116

- [«Installation du consignateur de base de données Java EE pour MFT», à la page 130](#)

Installation du consignateur de fichiers MFT autonome

Le consignateur de fichiers autonome est un processus Java qui doit se connecter à un gestionnaire de files d'attente de coordination à l'aide de liaisons IBM MQ . Pour définir un consignateur 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 consignateur de fichiers autonome, voir [«Configuration d'un consignateur MFT», à la page 107](#). Les étapes de cette rubrique permettent de configurer un consignateur pour qu'il se connecte à un gestionnaire de files d'attente de coordination. Pour d'autres configurations de consignateur, voir [«Autres configurations pour un consignateur MFT autonome», à la page 129](#)

Le consignateur de fichiers autonome n'est pas pris en charge sous z/OS ou 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 \(création d'un fichier MFT ou d'un consignateur de base de données\)», à la page 418](#).
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 la section [«Format du consignateur de fichiers MFT autonome», à la page 110](#).
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 la section [«Configuration de l'accès utilisateur pour un consignateur de fichiers MFT autonome», à la page 115](#).
6. Facultatif : Vous pouvez configurer 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 consignateur MFT», à la page 120](#).
7. **Windows**
Facultatif : Si vous utilisez un système Windows , vous pouvez exécuter le consignateur de fichiers autonome en tant que service Windows . Exécutez la commande **fteModifyLogger** avec le paramètre `-s` . Pour plus d'informations, voir la section [«fteModifyLogger \(exécutez un consignateur MFT en tant que service Windows \)», à la page 504](#).
8. Démarrez le consignateur de fichiers autonome à l'aide de la commande **fteStartLogger** . Pour plus d'informations, voir la section [«fteStartLogger», à la page 542](#).

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. Vous pouvez arrêter le consigneur à l'aide de la commande **fteStopLogger** . Pour plus d'informations, voir la section [«fteStopLogger»](#), à la page 546.

Résultats

Format du consigneur de fichiers MFT autonome

Le format des informations de message écrites par le consigneur de fichiers peut être défini dans le fichier `FileLoggerFormat.xml` .

Le répertoire de configuration du consigneur se trouve dans `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`. Lors de la création d'un nouveau consigneur de fichiers, une version de ce fichier est créée et contient un ensemble par défaut de définitions utilisées par le consigneur de fichiers. Pour plus d'informations sur la définition du format de journal par défaut, voir [«Format de journal par défaut du consigneur de fichiers autonome MFT»](#), à la page 629.

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;414d5120514d5f67627468696e6b20206466cf4e20004f02; [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 [«Format XSD du consigneur de fichier autonome»](#), à la page 634.

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 la section «Rubrique SYSTEM.FTE», à la page 636. 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 des messages du journal de transfert de fichiers», à la page 771](#)
- [«Formats de message du journal de transfert de fichiers planifiés», à la page 793](#)
- [«Format des messages du journal du moniteur MFT», à la page 798](#)

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 15. 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-Information	[SSIN]
Transfert scheduleSubmit	[SSTR]
scheduleSubmitTransferSet	[SSTS]
transferStarted	[TSTR]
transferCancelled	[TCAN]
transferComplete	[TCOM]
transferDelete	[TDEL]
transferProgress	[TPRO]

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>
```

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», à la page 629

Définition du format de fichier journal par défaut pour le consignateur de fichier Managed File Transfer autonome.

«Format XSD du consignateur de fichier autonome», à la page 634

Schéma d'un format de fichier autonome.

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 consigné 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 consignateur MFT](#)», à la page 120

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 consignateur

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 UNIX, 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 consignateur de fichiers autonomes, ajoutez les droits d'accès à ces noms de file d'attente à la place.

Installation du consignateur de base de données autonome MFT

Procédez comme suit pour installer et configurer le consignateur de base de données autonome.

Pourquoi et quand exécuter cette tâche

Important : Les programmes de consignment Managed File Transfer ne sont pas pris en charge sur la plateforme IBM i .

Pour plus d'informations sur le consignateur de base de données autonome, voir [«Configuration d'un consignateur MFT»](#), à la page 107.

Remarque : Vous ne pouvez pas exécuter simultanément plusieurs consignateurs 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 consignateur de base de données autonome. Pour plus d'informations, voir la section [«fteCreateLogger \(création d'un fichier MFT ou d'un consignateur de base de données\)»](#), à la page 418.
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 [«Propriétés de configuration du consignateur MFT»](#), à la page 120.
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.

 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 116. 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 consignateur MFT», à la page 120.
 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 la section «[fteModifyLogger \(exécutez un consignateur MFT en tant que service Windows\)](#)», à la page 504.
 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 `MQMFTCredentials.xsd`. Pour plus d'informations, voir la section «[Format de fichier de données d'identification MFT](#)», à la page 835. 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.

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 : 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/ftelog/  
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», à la page 120

Le consignateur Managed File Transfer possède un ensemble de propriétés de configuration. Spécifiez ces propriétés dans le fichier `logger.properties`, qui se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

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 consignateur de base de données autonome dépendent du nombre de systèmes que vous utilisez. Vous pouvez installer le consignateur de base de données autonome, IBM MQ et votre base de données sur un seul système ou sur deux systèmes. Le consignateur de base de données autonome doit se trouver sur le même système que IBM MQ. Les composants peuvent être installés dans les topologies suivantes:

Consignateur 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 consignateur de base de données autonome. Le consignateur 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.

Consignateur 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 `fte1log`, 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 `dspmqaout` ([display authority](#)) ou `dmpmqaout` ([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 UNIX, 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 `fte1og` ne fait partie d'aucun groupe d'administration Oracle (par exemple, `ora_dba` sous Windows ou `dba` sous UNIX)
- 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](#)

Propriétés de configuration du consignateur MFT

Le consignateur Managed File Transfer possède un ensemble de propriétés de configuration. Spécifiez ces propriétés dans le fichier `logger.properties`, qui se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

Pour IBM WebSphere MQ 7.5 ou version ultérieure, il est possible d'utiliser des variables d'environnement dans certaines propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction des changements d'environnement, tels que l'utilisateur qui exécute le processus. Pour plus d'informations, voir la section [«Utilisation des variables d'environnement dans les propriétés MFT»](#), à la page 547.

Remarque : Lorsque vous spécifiez des chemins de fichier sous Windows, le caractère de séparation barre oblique inversée (`\`) doit apparaître sous la forme de doubles barres obliques inversées (`\\`) (c'est-à-dire, avec caractères d'échappement `\\`). Vous pouvez également utiliser une barre oblique (`/`) comme séparateur. Pour plus d'informations sur l'échappement des caractères dans les fichiers de propriétés Java dans Oracle, voir [Javadoc pour la classe Properties](#).

Propriétés de connexion en mode liaisons

Nom de la propriété	Description	Valeur par défaut
<code>wmqfte.logger.type</code>	Type de consignateur utilisé: fichier ou base de données. Définissez cette valeur sur <code>FILE</code> ou <code>DATABASE</code> .	Aucune valeur par défaut
<code>wmqfte.max.transaction.messages</code>	Nombre maximal de messages traités dans une transaction avant la validation de la transaction. En mode de journalisation avec réutilisation automatique des journaux, un gestionnaire de files d'attente dispose d'une quantité fixe d'espace disponible pour les données en cours. Veillez à définir cette propriété avec une valeur suffisamment faible pour que l'espace disponible ne soit pas saturé.	50
<code>wmqfte.max.transaction.time</code>	Durée maximale, en millisecondes, qui s'écoule entre les validations de transaction.	5000
<code>wmqfte.max.consecutive.reject</code>	Nombre maximal de messages pouvant être rejetés consécutivement (c'est-à-dire sans rencontrer de message valide). Si ce nombre est dépassé, le consignateur conclut que le problème n'est pas lié aux messages eux-mêmes, mais à la configuration. Par exemple, si vous créez une colonne de nom d'agent dans la base de données plus étroite que tous les noms d'agent, tous les messages faisant référence aux agents sont rejetés.	50
<code>wmqfte.reject.queue.name</code>	Nom d'une file d'attente dans laquelle le consignateur insère des messages qu'il ne peut pas traiter. Si vous disposez d'un consignateur de base de données, voir Traitement et rejet des erreurs du consignateur de base de données pour plus de détails sur les messages pouvant être placés dans cette file d'attente.	SYSTEME SYSTEM.FTE.LOG.RJCT. <i>nom_connexion</i>

Tableau 16. Propriétés de connexion pour le mode liaisons (suite)

Nom de la propriété	Description	Valeur par défaut
wmqfte.command.queue.name	Nom d'une file d'attente à partir de laquelle le consignateur lit les messages de commande qui contrôlent son comportement.	SYSTEME SYSTEM.FTE.LOG.CMD.nom_journal
wmqfte.queue.manager	Gestionnaire de files d'attente auquel le consignateur se connecte. Ce paramètre est obligatoire et correspond à tout ce qui est nécessaire pour les connexions en mode liaison au gestionnaire de files d'attente. (Pour les propriétés de connexion à un gestionnaire de files d'attente éloignées, voir Tableau 17, à la page 128.)	Aucune valeur par défaut
wmqfte.message.source.type	<p>L'une de valeurs suivantes :</p> <p>Abonnement automatique Valeur par défaut. Le consignateur crée et utilise son propre abonnement durable et géré sur le gestionnaire de files d'attente défini dans SYSTEM.FTE/Log/#. Il s'agit d'une valeur appropriée pour la plupart des scénarios.</p> <p>abonnement d'administration Si l'abonnement automatique n'est pas approprié, vous pouvez définir un autre abonnement (par exemple, à l'aide de IBM MQ Explorer, MQSC ou PCF) et demander au consignateur d'utiliser cet abonnement. Par exemple, utilisez cette valeur pour partitionner l'espace journal de sorte qu'un consignateur gère les agents de A-H, un autre consignateur gère I-P et un troisième consignateur de Q-Z.</p> <p>file d'attente Si la topologie IBM MQ signifie que la création d'un abonnement pour le consignateur n'est pas pratique, vous pouvez utiliser une file d'attente à la place. Configurez IBM MQ de sorte que la file d'attente reçoive les messages qui sont généralement reçus par un abonnement à SYSTEM.FTE/Log/# sur le gestionnaire de file d'attente de coordination.</p>	Abonnement automatique
wmqfte.message.source.name	Si le type de source de messages est abonnement d'administration ou file d'attente, nom de l'abonnement ou de la file d'attente à utiliser. Cette propriété est ignorée si le type de source est abonnement automatique.	Aucune valeur par défaut
wmqfte.database.credentials.file	<p>Fichier contenant le nom d'utilisateur et le mot de passe pour la connexion à la base de données.</p> <p>Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.</p> <p>Pour plus d'informations, voir «Format de fichier de données d'identification MFT», à la page 835.</p>	<p> Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir Création d'un fichier de données d'identification MFT</p> <p> Pour plus d'informations sur l'emplacement et les droits d'accès de ce fichier, voir Configuration de MQMFTCredentials.xml .</p> <p> Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir MFT et IBM MQ authentification de la connexion .</p>

Tableau 16. Propriétés de connexion pour le mode liaisons (suite)

Nom de la propriété	Description	Valeur par défaut
<p>wmqfte.database.driver</p>	<p>Emplacement des classes du pilote JDBC pour la base de données. Il s'agit généralement du chemin et du nom d'un fichier JAR. Par exemple, le pilote de type 2 pour Db2 sur les systèmes AIX requiert le fichier /opt/IBM/db2/V9.5/java/db2jcc.jar. Sur les systèmes Windows, spécifiez le séparateur de chemin sous la forme d'une barre oblique (/), par exemple, C:/Program Files/IBM/SQLLIB/java/db2jcc.jar. z/OS Sous z/OS, indiquez le chemin d'accès complet du fichier db2jcc.jar. Par exemple, <code>wmqfte.database.driver=/db2/db2v10/jdbc/classes/db2jcc.jar</code>.</p> <p>z/OS Sur les systèmes z/OS, vous devez référencer tous les fichiers JAR suivants:</p> <ul style="list-style-type: none"> • db2jcc.jar • db2jcc_license_cisuz.jar • db2jcc_javax.jar <p>Si votre pilote de base de données se compose de plusieurs fichiers JAR (par exemple, Db2 9.1 requiert un fichier JAR de pilote et un fichier JAR de licence), incluez tous ces fichiers JAR dans cette propriété. Séparez plusieurs noms de fichier à l'aide du séparateur de chemin d'accès aux classes de votre plateforme, c'est-à-dire le point-virgule (;) sur les systèmes Windows et le signe deux-points (:) sur les autres plateformes.</p>	<p>Aucune valeur par défaut</p>
<p>wmqfte.database.exclude_metadata</p>	<p>Contrôle si les entrées sont stockées dans la table de métadonnées qui contient des informations pouvant être trouvées dans d'autres tables du schéma du consignateur de base de données. Définissez cette valeur sur <code>true</code> ou sur <code>false</code>. Ces entrées de métadonnées ne sont plus stockées par défaut car il s'agit d'une duplication des données existantes et d'un gaspillage de la capacité de stockage de la base de données. Les entrées de propriété et les tables dans lesquelles les mêmes données apparaissent sont les suivantes:</p> <ul style="list-style-type: none"> • com.ibm.wmqfte.SourceAgent TRANSFER_EVENT ou CALL_REQUEST • com.ibm.wmqfte.DestinationAgent TRANSFER_EVENT • com.ibm.wmqfte.MqmdUser TRANSFER_EVENT ou CALL_REQUEST • com.ibm.wmqfte.OriginatingUser TRANSFER_EVENT ou CALL_REQUEST • com.ibm.wmqfte.OriginatingHost TRANSFER_EVENT ou CALL_REQUEST • com.ibm.wmqfte.TransferId TRANSFER ou CALL_REQUEST • com.ibm.wmqfte.JobName TRANSFER ou CALL_REQUEST <p>La définition de la valeur de cette propriété sur <code>false</code> entraîne le stockage de ces entrées de métadonnées dans la table de métadonnées.</p>	<p>conforme</p>

Tableau 16. Propriétés de connexion pour le mode liaisons (suite)

Nom de la propriété	Description	Valeur par défaut
wmqfte.database.host	<p>Db2 uniquement :</p> <p>Pour IBM WebSphere MQ 7.5 ou version ultérieure, nom d'hôte du serveur de base de données auquel se connecter à l'aide d'un pilote JDBC de type 4. Si une valeur est spécifiée pour cette propriété, une valeur pour <code>wmqfte.database.port</code> doit également être spécifiée. Si les deux propriétés ne sont pas définies, le consigneur de base de données se connecte à l'aide du pilote JDBC de type 2 par défaut.</p> <p>Si une valeur est spécifiée pour cette propriété, un fichier de données d'identification pour ce consigneur (chemin d'accès au fichier défini par la propriété <code>wmqfte.database.credentials.file</code>) doit exister et être accessible pour définir le nom d'utilisateur et le mot de passe de connexion à la base de données, même si la base de données se trouve sur le système local.</p>	Aucune valeur par défaut
wmqfte.database.name	Nom de l'instance de base de données (ou du sous-système lors de l'utilisation de Db2 for z/OS) qui contient les tables de journal Managed File Transfer .	Aucune valeur par défaut
wmqfte.database.type	Système de gestion de base de données en cours d'utilisation: Db2 ou Oracle. Définissez cette valeur sur <code>db2</code> ou <code>oracle</code> .	db2
wmqfte.database.port	<p>Db2 uniquement :</p> <p>Pour IBM WebSphere MQ 7.5 ou version ultérieure, numéro de port du serveur de base de données auquel se connecter à l'aide d'un pilote JDBC de type 4. Si une valeur est spécifiée pour cette propriété, une valeur pour <code>wmqfte.database.host</code> doit également être spécifiée. Si les deux propriétés ne sont pas définies, le consigneur de base de données se connecte à l'aide du pilote JDBC de type 2 par défaut.</p> <p>Si une valeur est spécifiée pour cette propriété, un fichier de données d'identification pour ce consigneur (chemin d'accès au fichier défini par la propriété <code>wmqfte.database.credentials.file</code>) doit exister et être accessible pour définir le nom d'utilisateur et le mot de passe de connexion à la base de données, même si la base de données se trouve sur le système local.</p>	Aucune valeur par défaut
wmqfte.database.schema	Schéma de base de données qui contient les tables de journalisation Managed File Transfer . Dans la plupart des cas, la valeur par défaut est appropriée, mais vous devrez peut-être spécifier une autre valeur en fonction de vos propres considérations de base de données spécifiques au site.	FTELOG
wmqfte.database.native.library.path	<p>Chemin contenant les bibliothèques natives requises par le pilote de base de données de votre choix (le cas échéant). Par exemple, le pilote de type 2 pour Db2 sur les systèmes AIX requiert des bibliothèques de <code>/opt/IBM/db2/V9.5/lib32/</code>. Comme alternative à cette propriété, vous pouvez définir la propriété système <code>java.library.path</code> à l'aide d'autres méthodes.</p> <p>Sur les systèmes Solaris et HP-UX , avant d'exécuter la commande fteStartLogger , vous devez également définir et exporter la variable d'environnement <code>LD_LIBRARY_PATH</code> pour inclure le chemin.</p>	Aucune valeur par défaut

Tableau 16. Propriétés de connexion pour le mode liaisons (suite)

Nom de la propriété	Description	Valeur par défaut
wmqfte.file.logger.fileDirectory	Répertoire dans lequel se trouvent les fichiers journaux du consignateur de fichiers.	mqft/logs/coordination_dir/loggers/logger_name/logs
wmqfte.file.logger.fileSize	Taille maximale qu'un fichier journal peut atteindre. La valeur de taille est un entier positif, supérieur à zéro, suivi de l'une des unités suivantes: Ko, Mo, Go, m (minutes), h (heures), d (jours), w (semaines). Par exemple, <code>wmqfte.file.logger.fileSize=5MB</code> Spécifie une taille de fichier maximale de 5MB. <code>wmqfte.file.logger.fileSize=2d</code> Indique une taille de fichier maximale de 2 jours de données.	10MB
wmqfte.file.logger.fileCount	Nombre maximal de fichiers journaux à créer. Lorsque la quantité de données dépasse la quantité maximale pouvant être stockée dans ce nombre de fichiers, le fichier le plus ancien est supprimé de sorte que le nombre de fichiers ne dépasse jamais la valeur spécifiée.	3

Tableau 16. Propriétés de connexion pour le mode liaisons (suite)

Nom de la propriété	Description	Valeur par défaut
<p>wmqfte.file.logger.mode</p>	<p>Mode de consignateur utilisé: circulaire ou linéaire. Définissez cette valeur sur CIRCULAIRE ou sur LINEAR.</p> <p>CIRCULAIRE-Le consignateur de fichiers écrit des informations dans un fichier jusqu'à ce que ce dernier atteigne sa taille maximale définie à l'aide de la propriété wmqfte.file.logger.fileSize. Lorsque la taille maximale est atteinte, le consignateur de fichiers démarre un nouveau fichier. Le nombre maximal de fichiers écrits dans ce mode est contrôlé par la valeur définie à l'aide de la propriété wmqfte.file.logger.fileCount. Lorsque ce nombre maximal de fichiers est atteint, le consignateur de fichiers supprime le premier fichier et le recrée pour l'utiliser comme fichier actif. Si la valeur définie dans la propriété wmqfte.file.logger.fileSize est une unité d'octet de taille fixe (par exemple, Ko, Mo ou Go), la limite supérieure de l'espace disque utilisé dans ce mode est égale à fileSize multipliée par fileCount. Si la valeur définie dans la propriété wmqfte.file.logger.fileSize est une unité de temps (par exemple, m, h, dou w), la taille maximale dépend du débit des messages de journal dans votre système au cours de ces périodes. La convention de dénomination des fichiers journaux utilisée lors de l'exécution dans ce mode est: <i>logger_namenumber-timestamp</i>.log où:</p> <ul style="list-style-type: none"> • <i>logger_name</i> est le nom donné au consignateur dans la commande fteCreateLogger. • <i>nombre</i> est le numéro du fichier dans l'ensemble. • <i>horodatage</i> est l'horodatage de la création du fichier. <p>Exemple: LOGGER1-20111216123430147.log</p> <p>LINEAR-Le consignateur de fichiers écrit des informations dans un fichier jusqu'à ce que ce fichier atteigne sa taille maximale définie à l'aide de la propriété wmqfte.file.logger.fileSize. Lorsque la taille maximale est atteinte, le consignateur de fichiers démarre un nouveau fichier. Les fichiers précédemment écrits ne sont pas supprimés, ce qui permet de les conserver en tant qu'enregistrement historique des messages de journal. Les fichiers ne sont pas supprimés lors de l'exécution en mode linéaire. Par conséquent, la propriété wmqfte.file.logger.fileCount est ignorée car il n'y a pas de limite supérieure au nombre de fichiers pouvant être créés. Etant donné qu'il n'y a pas de limite supérieure lors de l'exécution dans ce mode, il est nécessaire de suivre la quantité d'espace disque utilisée par les fichiers journaux afin d'éviter une faible quantité d'espace disque. La convention de dénomination des fichiers journaux utilisée lors de l'exécution dans ce mode est: <i>logger_name-timestamp</i>.log où:</p> <ul style="list-style-type: none"> • <i>logger_name</i> est le nom donné au consignateur dans la commande fteCreateLogger. • <i>horodatage</i> est l'horodatage de la création du fichier. <p>Exemple: LOGGER-20111216123430147.log</p>	<p>Aucune valeur par défaut</p>

Tableau 16. Propriétés de connexion pour le mode liaisons (suite)

Nom de la propriété	Description	Valeur par défaut
wmqfte.max.retry.interval	<p>Durée maximale, en secondes, entre les nouvelles tentatives lorsque le consignateur rencontre une erreur persistante.</p> <p>Certaines conditions d'erreur (par exemple, la perte de la connexion à la base de données) empêchent le consignateur de continuer. Lorsque ce type de condition se produit, le consignateur annule la transaction en cours, attend un certain temps, puis effectue de nouvelles tentatives. Le temps d'attente du consignateur est initialement très court, de sorte que les erreurs transitoires peuvent être surmontées rapidement. Toutefois, à chaque nouvelle tentative du consignateur, le temps d'attente est augmenté. Cela évite trop de travail inutile lorsque la condition d'erreur est plus longue, par exemple lorsqu'une base de données est arrêtée à des fins de maintenance.</p> <p>Utilisez cette propriété pour définir une limite à la durée de l'attente de sorte qu'une nouvelle tentative se produise dans un délai raisonnable après la résolution de la condition d'erreur.</p>	600
loggerQMgrRetryInterval	Intervalle, en secondes, entre les vérifications de la disponibilité du gestionnaire de files d'attente par le contrôleur de processus du consignateur.	30
maxRestartCount	Nombre maximal de redémarrages pouvant se produire dans l'intervalle de temps spécifié par la valeur de la propriété d'intervalle maxRestart. Lorsque cette valeur est dépassée, le contrôleur de processus du consignateur arrête de redémarrer le consignateur et effectue à la place une action basée sur la valeur de la propriété de délai maxRestart.	4
maxRestartInterval	Intervalle, en secondes, pendant lequel le contrôleur de processus du consignateur mesure les redémarrages du consignateur. Si le nombre de redémarrages dans cet intervalle dépasse la valeur de la propriété Nombre maxRestart, le contrôleur de processus du consignateur arrête de redémarrer le consignateur. A la place, le contrôleur de processus du consignateur effectue une action basée sur la valeur de la propriété de délai maxRestart.	130
maxRestartDelay	Détermine le comportement du contrôleur de processus du consignateur lorsque le taux de redémarrages du consignateur dépasse la valeur des propriétés d'intervalle maxRestart et maxRestart. Si vous indiquez une valeur inférieure ou égale à zéro, le contrôleur de processus du consignateur est arrêté. Si vous indiquez une valeur supérieure à zéro, il s'agit du nombre de secondes à attendre avant que les informations d'historique de redémarrage détenues par le contrôleur de processus du consignateur soient réinitialisées et que le consignateur soit redémarré.	-1
wmqfte.oracle.port	Port utilisé par le consignateur pour se connecter à l'instance Oracle . Ce port est également appelé programme d'écoute TNS.	1521
wmqfte.oracle.host	Hôte utilisé par le consignateur pour se connecter à l'instance Oracle .	système hôte local

Tableau 16. Propriétés de connexion pour le mode liaisons (suite)

Nom de la propriété	Description	Valeur par défaut
armELEMTYPE	Propriété facultative. Si le consignateur est configuré pour être redémarré par le gestionnaire de redémarrage automatique (ARM), définissez cette propriété sur la valeur du paramètre ARM ELEMTYPE spécifiée dans la règle ARM associée. Pour un consignateur, définissez ELEMTYPE sur SYSBFGLG.	Non définie
armELEMENT	Propriété facultative. Si le consignateur est configuré pour être redémarré par le gestionnaire de redémarrage automatique (ARM), définissez cette propriété sur la valeur du paramètre ARM ELEMENT spécifiée dans la règle ARM associée. Vous pouvez définir la valeur ELEMENT pour qu'elle corresponde au nom du consignateur.	Non définie
loggerQMgrAuthenticationCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification de connexion MQ pour la connexion au gestionnaire de files d'attente de coordination du consignateur.	 Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir Création d'un fichier de données d'identification MFT  Pour plus d'informations sur l'emplacement et les droits d'accès de ce fichier, voir Configuration de MQMFTCredentials.xml .  Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir MFT et IBM MQ authentification de la connexion .
trace	<p>Propriété facultative. Spécification de trace lorsque le consignateur doit être exécuté avec la trace activée au démarrage du consignateur. La spécification de trace est une liste de classes séparée par des virgules, suivie du signe égal et d'un niveau de trace.</p> <p>Par exemple, <code>com.ibm.wmqfte.databaselogger,com.ibm.wmqfte.databaselogger.operation=all</code>.</p> <p>Vous pouvez spécifier plusieurs spécifications de trace dans une liste de valeurs séparées par deux-points. Par exemple <code>:com.ibm.wmqfte.databaselogger=moderate:com.ibm.wmqfte.databaselogger.operation=all</code></p>	Aucun
traceFiles	Propriété facultative. Nombre total de fichiers de trace à conserver. Cette valeur s'applique au contrôleur de processus d'un consignateur, ainsi qu'au consignateur lui-même.	5
traceSize	Propriété facultative. Taille maximale en Mo de chaque fichier de trace, avant que la trace passe au fichier suivant. Cette valeur s'applique au contrôleur de processus du consignateur et au consignateur lui-même.	20

Propriétés de connexion en mode client

V 9.0.4

Le tableau suivant présente les propriétés de consignateur supplémentaires requises pour la prise en charge de la connexion en mode client à un gestionnaire de files d'attente de consignateur.

Tableau 17. Propriétés de connexion pour le mode client

Nom de la propriété	Description	Valeur par défaut
wmqfte.queue.manager.host	Nom d'hôte ou adresse IP du gestionnaire de files d'attente du consignateur.	Aucune valeur par défaut
wmqfte.queue.manager.port	Port sur lequel le gestionnaire de files d'attente du consignateur est en mode écoute.	1414
wmqfte.queue.manager.channel	Nom du canal de connexion serveur sur le gestionnaire de files d'attente du consignateur.	SYSTEM.DEF.SVRCONN
wmqfte.Ssl.CipherSuite	Indique les aspects TLS de la façon dont le consignateur et le gestionnaire de files d'attente du consignateur échangent des données. La valeur de wmqfte.Ssl.CipherSuite est un nom CipherSuite . Le nom CipherSuite est mappé au nom CipherSpec utilisé sur le canal du gestionnaire de files d'attente du consignateur. Pour plus d'informations, voir CipherSuite et CipherSpec name mappings .	Aucune valeur par défaut
wmqfte.Ssl.PeerName	Indique un squelette de nom distinctif qui doit correspondre au nom fourni par le gestionnaire de files d'attente du consignateur. Le nom distinctif est utilisé pour vérifier le certificat d'identification présenté par le gestionnaire de files d'attente lors de la connexion.	Aucune valeur par défaut
wmqfte.Ssl.TrustStore	Indique l'emplacement des certificats auxquels le consignateur fait confiance. La valeur de wmqfte.Ssl.TrustStore est un chemin de fichier. Si le chemin d'accès au fichier est un chemin d'accès au fichier Windows , la barre oblique inversée (\) doit être mise en échappement avec une autre barre oblique inversée (\\). Notez que la valeur de cette propriété peut contenir des variables d'environnement.	Aucune valeur par défaut
wmqfte.Ssl.TrustStoreCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification wmqfte.Ssl.TrustStore . Notez que la valeur de cette propriété peut contenir des variables d'environnement.	Aucune valeur par défaut
wmqfte.Ssl.TrustStoreType	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être jks ou pkcs12.	jks
wmqfte.Ssl.KeyStore	Indique l'emplacement de la clé privée du consignateur. La valeur de wmqfte.Ssl.KeyStore est un chemin de fichier. Si le chemin d'accès au fichier est un chemin d'accès au fichier Windows , la barre oblique inversée (\) doit être mise en échappement avec une autre barre oblique inversée (\\). Notez que la valeur de cette propriété peut contenir des variables d'environnement.	Aucune valeur par défaut

Tableau 17. Propriétés de connexion pour le mode client (suite)

Nom de la propriété	Description	Valeur par défaut
wmqfte.Ssl.KeyStore.CredentialsFile	Chemin d'accès au fichier qui contient les données d'identification wmqfte.Ssl.KeyStore . Notez que la valeur de cette propriété peut contenir des variables d'environnement.	Aucune valeur par défaut
wmqfte.Ssl.KeyStoreType	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être jks ou pkcs12.	jks
wmqfte.Ssl.FipsRequired	Indique que vous souhaitez activer la prise en charge de FIPS au niveau du consignateur. La valeur de cette propriété peut être true ou false. Pour plus d'informations, voir «Prise en charge de FIPS dans MFT», à la page 688.	false

Référence associée

«Utilisation des variables d'environnement dans les propriétés MFT», à la page 547

Depuis la IBM WebSphere MQ 7.5, il est possible d'utiliser des variables d'environnement dans les propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction des changements d'environnement. Par exemple, l'utilisateur qui exécute le processus.

«Propriétés SSL pour MFT», à la page 622

Utilisez SSL ou TLS avec IBM MQ et Managed File Transfer pour empêcher les connexions non autorisées entre les agents et les gestionnaires de files d'attente et pour chiffrer le trafic des messages entre les agents et les gestionnaires de files d'attente.

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 du consignateur de base de données](#).

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 consigneateur 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 consigneateur. 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 consigneateur 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 consigneateur autonome génère le message d'erreur BFGDB0002E. Ce message d'erreur implique un problème avec la configuration du consigneateur autonome.

File d'attente

La topologie typique est celle où le consigneateur 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 consigneateur 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 consigneateur 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 consigneateur 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 consigneateur de base de données Java EE pour MFT

Suivez ces instructions pour installer et configurer le consigneateur de base de données Java EE à utiliser avec Managed File Transfer.

Pourquoi et quand exécuter cette tâche

Pour plus d'informations sur le consigneateur de base de données Java EE, voir la rubrique [«Configuration d'un consigneateur MFT»](#), à la page 107.

Remarque : Vous ne pouvez pas exécuter un consigneateur de base de données Java EE en même temps qu'un consigneateur autonome, sauf si ces consigneateurs utilisent des instances distinctes de la base de données.

Procédure

1. Avant d'installer le consigneur de base de données Java EE , vous devez préparer votre environnement. Utilisez les instructions de la rubrique [«Préparation de l'installation du consigneur de base de données Java EE pour MFT»](#), à la page 131.
2. Vous installez le consigneur de base de données Java EE sur un serveur d'applications compatible avec Java Platform, Enterprise Edition (Java EE). Pour connaître les instructions, consultez les rubriques suivantes :
 - [«Installation du consigneur de base de données Java EE pour MFT avec WebSphere Application Server 7.0»](#), à la page 134
 - [«Installation du consigneur de base de données Java EE pour MFT avec WebSphere Application Server Community Edition»](#), à la page 139

Tâches associées

[«Préparation de l'installation du consigneur de base de données Java EE pour MFT»](#), à la page 131
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 7.0»](#), à la page 134

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 7.0.

[«Installation du consigneur de base de données Java EE pour MFT avec WebSphere Application Server Community Edition»](#), à la page 139

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 Community Edition.

[«Configuration de l'accès utilisateur pour le consigneur de base de données Java EE pour MFT»](#), à la page 143

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 145

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 Java EE . 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 JEE. Sauvegardez votre base de données avant la migration.

Référence associée

[«Droits d'accès au consigneur MFT»](#), à la page 372

L'utilisateur du système d'exploitation qui exécute le consigneur requiert certains droits IBM MQ sur les files d'attente du consigneur et sur SYSTEM.FTE .

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 107.

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.
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 `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. 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 la section «[Modification du nom de schéma dans votre consignateur de base de données Java EE pour MFT](#)», à la page 132.
 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 JEE](#).

Que faire ensuite

Vous pouvez maintenant installer le consignateur de base de données Java EE sur un serveur d'applications compatible avec Java EE. Utilisez les instructions des rubriques suivantes, en fonction du serveur d'applications que vous utilisez:

- «[Installation du consignateur de base de données Java EE pour MFT avec WebSphere Application Server 7.0](#)», à la page 134
- «[Installation du consignateur de base de données Java EE pour MFT avec WebSphere Application Server Community Edition](#)», à la page 139

Modification du nom de schéma dans votre 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 votre 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 `persistance.xml` du fichier JAR JPA à l'aide de la commande suivante:

```
jar -xvf 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.

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 7.0», à la page 134

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 7.0.

[«Installation du consignateur de base de données Java EE pour MFT avec WebSphere Application Server Community Edition», à la page 139](#)

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 Community Edition.

Définition du chemin d'accès à la bibliothèque native dans WebSphere Application Server 7.0

Si vous déployez l'application de consignateur de base de données Java Platform, Enterprise Edition (Java EE) sous WebSphere Application Server 7.0 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 7.0 :

```
A connection could not be made to WebSphere MQ for the following reason:  
CC=2;RC=2495;AMQ8568: The native JNI library 'mqjbn0' was not found. [3=mqjbn0]
```

Utilisez la console d'administration WebSphere Application Server 7.0 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**.
5. Redémarrez le serveur d'applications pour actualiser la configuration.
6. Obligatoire : Redémarrez le serveur d'applications une deuxième fois pour charger les bibliothèques.

Installation du consignateur de base de données Java EE pour MFT avec WebSphere Application Server 7.0

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 7.0.

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 131](#) et [«Définition du chemin d'accès à la bibliothèque native dans WebSphere Application Server 7.0», à la page 134](#).

Pourquoi et quand exécuter cette tâche

Pour plus d'informations sur le consigneur de base de données Java EE , voir «[Configuration d'un consigneur MFT](#)», à la page 107.

Procédure

1. Configurez le fournisseur JDBC XA:
 - a) Sélectionnez **Ressources > JDBC > JDBC Fournisseurs** dans la navigation de la console d'administration WebSphere Application Server 7.0 .
 - 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**.
 - 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 7.0 .
 - 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 consigneur de base de données JEE](#).
 - 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 7.0 .
 - 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 à 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 7.0 .
 - b) Cliquez sur le bouton **Tester connexion**.
5. Créez une rubrique.
- a) Dans la navigation de la console d'administration WebSphere Application Server 7.0 , 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 7.0 , 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) Cliquer 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 consignateur 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 définie sur plus de 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 la section [«Traitement et rejet des erreurs du consignateur MFT»](#), à la page 339.

7. Créez une fabrique de connexions de file d'attente.

- a) Dans la navigation de la console d'administration WebSphere Application Server 7.0, cliquez sur **Ressources > JMS > Fabriques de connexions de file d'attente**.
- 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 2, sélectionnez **Entrez toutes les informations requises dans cet assistant**.
- g) 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**.
- h) 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**.
- i) 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.
- j) 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:

- a) Dans la navigation de la console d'administration WebSphere Application Server 7.0, cliquez sur **Ressources > JMS > Files d'attente**.

- 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) Choisissez des valeurs uniques pour les zones **Nom** et **Nom JNDI**, que vous référencerez à nouveau ultérieurement dans la configuration.
 - f) 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 7.0, 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 7.0 :
- 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**.

Installation du consigneur de base de données Java EE pour MFT avec WebSphere Application Server Community Edition

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 Community Edition.

Avant de commencer

Avant d'installer l'application de consigneur de base de données Java EE , suivez les instructions de la rubrique [«Préparation de l'installation du consigneur de base de données Java EE pour MFT»](#), à la page 131.

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 107.

Procédure

1. Déployez l'adaptateur de ressources IBM MQ , `wmq.jmsra.rar`.
 - Pour déployer l'adaptateur de ressources IBM MQ pour un consigneur de base de données JEE à l'aide d'un gestionnaire de files d'attente de coordination QM_JUPITER, procédez comme suit. Cet exemple s'applique lorsque votre instance WebSphere Application Server Community Edition s'exécute sur le même système que le gestionnaire de files d'attente IBM MQ auquel vous souhaitez vous connecter.
 - a. Créez un fichier de plan qui définit une connexion au gestionnaire de files d'attente de coordination MFT . L'exemple de fichier de plan suivant définit une connexion à un gestionnaire de files d'attente appelé QM_JUPITER et une référence à une file d'attente appelée SYSTEM.FTE.LOG.RJCT.LOGGER1 sur ce gestionnaire de files d'attente.

```
<?xml version="1.0" encoding="UTF-8"?>
<connector xmlns="http://geronimo.apache.org/xml/ns/j2ee/connector">
  <resourceadapter>
    <resourceadapter-instance>
      <resourceadapter-name>WMQ</resourceadapter-name>
      <workmanager>
        <gbean-link>DefaultWorkManager</gbean-link>
      </workmanager>
    </resourceadapter-instance>
  </resourceadapter>
  <outbound-resourceadapter>
    <connection-definition>
      <connectionfactory-interface>javax.jms.ConnectionFactory</connectionfactory-interface>
      <connectiondefinition-instance>
        <name>jms/WMQFTEJEEDBLoggerRejectQueueCF</name>
        <config-property-setting name="queueManager">QM_JUPITER</config-property-setting>
        <config-property-setting name="transportType">BINDINGS</config-property-setting>
      </connectiondefinition-instance>
    </connection-definition>
  </outbound-resourceadapter>
</connector>
```

```

<xa-transaction>
  <transaction-caching/>
</xa-transaction>
<single-pool>
  <max-size>10</max-size>
  <min-size>1</min-size>
  <blocking-timeout-milliseconds>5000</blocking-timeout-milliseconds>
  <idle-timeout-minutes>2</idle-timeout-minutes>
  <match-all />
</single-pool>
</connectionmanager>
</connectiondefinition-instance>
</connection-definition>
</outbound-resourceadapter>
</resourceadapter>
<adminobject>
  <adminobject-interface>javax.jms.Queue</adminobject-interface>
  <adminobject-class>com.ibm.mq.connector.outbound.MQQueueProxy</adminobject-class>
  <adminobject-instance>
    <message-destination-name>jms/WMQFTEJEEEDBLoggerRejectQueue</message-destination-name>
    <config-property-setting name="baseQueueManagerName">QM_JUPITER</config-property-setting>
    <config-property-setting name="baseQueueName">SYSTEM.FTE.LOG.RJCT.LOGGER1</config-property-setting>
  </adminobject-instance>
</adminobject>
</connector>

```

Pour utiliser ce fichier de plan dans votre environnement, remplacez QM_JUPITER par le nom de votre gestionnaire de files d'attente de coordination.

- b. Ouvrez la console d'administration WebSphere Application Server CE.
 - c. Dans la liste **Actions de la console commune** de la **page d'accueil**, cliquez sur **Déployer de nouvelles applications > Déployer de nouvelles applications**.
 - d. Dans la zone **Archive**, entrez `mq_install_root/java/lib/jca/wmq.jmsra.rar`
 - e. Dans la zone **Plan**, entrez le chemin d'accès au fichier de plan que vous avez créé à l'étape 1a.
- Si votre instance WebSphere Application Server Community Edition s'exécute sur un système différent du gestionnaire de files d'attente IBM MQ auquel vous souhaitez vous connecter, procédez comme suit pour déployer l'adaptateur de ressources IBM MQ.
 - a. Créez un fichier de plan qui définit une connexion au gestionnaire de files d'attente de coordination WMQFTE. L'exemple de fichier de plan suivant définit une connexion à un gestionnaire de files d'attente, QM_SATURN, qui se trouve sur un système différent de celui de votre installation WebSphere Application Server Community Edition, et une référence à une file d'attente appelée SYSTEM.FTE.LOG.RJCT.LOGGER1 sur ce gestionnaire de files d'attente. Le nom d'hôte de QM_SATURN est saturn.example.com. Le port de QM_SATURN est 1415. Le canal de QM_SATURN est SYSTEM.DEF.SVRCONN.

Etant donné que le serveur d'applications et le gestionnaire de files d'attente se trouvent sur des systèmes différents, vous devez utiliser une connexion en mode client au gestionnaire de files d'attente. Le fichier de plan suivant définit la valeur de l'élément `<config-property-setting>` dont le nom est `transportType` sur `CLIENT`.

```

<?xml version="1.0" encoding="UTF-8"?>
<connector xmlns="http://geronimo.apache.org/xml/ns/j2ee/connector">
  <resourceadapter>
    <resourceadapter-instance>
      <resourceadapter-name>WMQ</resourceadapter-name>
      <workmanager>
        <gbean-link>DefaultWorkManager</gbean-link>
      </workmanager>
    </resourceadapter-instance>
  <outbound-resourceadapter>
    <connection-definition>
      <connectionfactory-interface>javax.jms.ConnectionFactory</connectionfactory-interface>
      <connectiondefinition-instance>
        <name>jms/WMQFTEJEEEDBLoggerRejectQueueCF</name>
        <config-property-setting name="queueManager">QM_SATURN</config-property-setting>
        <config-property-setting name="transportType">CLIENT</config-property-setting>
        <config-property-setting name="channel">SYSTEM.DEF.SVRCONN</config-property-setting>
        <config-property-setting name="hostName">saturn.example.com</config-property-setting>
        <config-property-setting name="port">1415</config-property-setting>
      </connectiondefinition-instance>
    </connection-definition>
  </outbound-resourceadapter>
</connector>

```

```

<xa-transaction>
  <transaction-caching/>
</xa-transaction>
<single-pool>
  <max-size>10</max-size>
  <min-size>1</min-size>
  <blocking-timeout-milliseconds>5000</blocking-timeout-milliseconds>
  <idle-timeout-minutes>2</idle-timeout-minutes>
  <match-all />
</single-pool>
</connectionmanager>
</connectiondefinition-instance>
</connection-definition>
</outbound-resourceadapter>
</resourceadapter>
<adminobject>
  <adminobject-interface>javax.jms.Queue</adminobject-interface>
  <adminobject-class>com.ibm.mq.connector.outbound.MQQueueProxy</adminobject-class>
  <adminobject-instance>
    <message-destination-name>jms/WMQFTEJEEEDBLoggerRejectQueue</message-destination-name>
    <config-property-setting name="baseQueueManagerName">QM_SATURN</config-property-setting>
    <config-property-setting name="baseQueueName">SYSTEM.FTE.LOG.RJCT.LOGGER1</config-property-setting>
  </adminobject-instance>
</adminobject>
</connector>

```

Pour utiliser ce fichier de plan dans votre environnement, remplacez QM_SATURN par le nom de votre gestionnaire de files d'attente de coordination. Remplacez la valeur du nom d'hôte, du port et du canal par les valeurs de votre gestionnaire de files d'attente de coordination.

- b. Copiez le fichier `mq_install_root/java/lib/jca/wmq.jmsra.rar` du système où IBM MQ est installé vers le système où WebSphere Application Server CE est installé.
- c. Ouvrez la console d'administration WebSphere Application Server CE.
- d. Dans la liste **Actions de la console commune** de la **page d'accueil**, cliquez sur **Déployer de nouvelles applications > Déployer de nouvelles applications**.
- e. Dans la zone **Archive**, entrez le chemin d'accès à la copie du fichier `wmq.jmsra.rar` que vous avez obtenue.
- f. Dans la zone **Plan**, entrez le chemin d'accès au fichier de plan que vous avez créé.

2. Vous devez définir un connecteur de base de données pour que l'application de consignateur de base de données JEE puisse accéder à la base de données requise à partir de l'environnement WebSphere Application Server Community Edition.

Effectuez les étapes suivantes à partir de la console d'administration WebSphere Application Server Community Edition :

- a) Selon le niveau de WebSphere Application Server Community Edition que vous utilisez, dans la **Navigations dans la console**, sélectionnez **Services > Pools de bases de données** ou sélectionnez **Ressources > Sources de données**.
- b) Créez un pool de bases de données à l'aide de l'assistant de pool de bases de données Geronimo. Dans la zone **Nom du pool de bases de données**, entrez `jdbc/wmqfte-database`.
- c) Pour **Type de base de données**, sélectionnez `DB2 XA` ou `Oracle Thin`, en fonction de votre base de données.
- d) Cliquez sur **Suivant**.
- e) Dans la zone **Driver jar**, sélectionnez le fichier jar approprié pour votre base de données.
- f) Dans la zone **Nom de la base de données**, entrez le nom de la base de données à laquelle vous vous connectez pour les informations de statut de transfert.
- g) Dans la zone **Nom d'utilisateur**, entrez le nom d'utilisateur pour la connexion et l'authentification avec votre base de données.
- h) Dans les zones **Mot de passe** et **Confirmation du mot de passe**, entrez le mot de passe pour l'authentification auprès de votre base de données.
- i) Dans la zone **Numéro de port**, entrez le numéro de port que vous utilisez s'il ne s'agit pas du port par défaut.

- j) Vérifiez que la valeur de **Type de pilote** est 4.
 - k) Sélectionnez XA dans la liste **Type de transaction** .
 - l) Cliquez sur **Déployer**.
3. Mettez à jour le fichier `openejb-jar.xml` de l'application de consignateur de base de données Managed File Transfer JEE pour votre environnement. Utilisez un utilitaire Java SDK `jar` pour effectuer les étapes suivantes:

- a) Extrayez le fichier JAR d'EJB du fichier EAR fourni en exécutant la commande suivante:

```
jar -xf ear_file_name com.ibm.wmqfte.databaselogger.jee.ejb.jar
```

où *nom_fichier_ear* est `com.ibm.wmqfte.databaselogger.jee.ear` ou `com.ibm.wmqfte.databaselogger.jee.oracle.ear` selon que vous utilisez Db2 ou Oracle. Le fichier EAR se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/web` de l'installation de IBM WebSphere MQ File Transfer Edition Server.

- b) Extrayez le fichier META-INF/`openejb-jar.xml` du fichier JAR d'EJB précédemment extrait, `com.ibm.wmqfte.databaselogger.jee.ejb.jar`, en exécutant la commande suivante:

```
jar -xf com.ibm.wmqfte.databaselogger.jee.ejb.jar META-INF/openejb-jar.xml
```

- c) Utilisez un éditeur de texte pour éditer le fichier META-INF/`openejb-jar.xml` extrait. Modifiez les valeurs `activation-config-property` suivantes pour qu'elles correspondent à votre environnement:

queueManager

Nom du gestionnaire de files d'attente IBM MQ utilisé par le consignateur de base de données JEE.

hostName

Nom d'hôte à utiliser pour la connexion au gestionnaire de files d'attente IBM MQ spécifié. Cette valeur n'est pas requise si vous vous connectez au gestionnaire de files d'attente en mode liaisons.

transportType

Indique s'il faut se connecter au gestionnaire de files d'attente IBM MQ spécifié en mode client ou liaisons.

port

Non obligatoire si vous avez spécifié un **transportType** de liaisons. Port à utiliser pour la connexion au gestionnaire de files d'attente IBM MQ spécifié.

canal

Non obligatoire si vous avez spécifié un **transportType** de liaisons. Canal serveur à utiliser pour la connexion au gestionnaire de files d'attente IBM MQ spécifié.

- d) Mettez à jour le fichier JAR d'EJB avec le fichier META-INF/`openejb-jar.xml` modifié en exécutant la commande suivante:

```
jar -uf com.ibm.wmqfte.databaselogger.jee.ejb.jar META-INF/openejb-jar.xml
```

- e) Mettez à jour le fichier EAR fourni avec le fichier JAR d'EJB mis à jour en exécutant la commande suivante:

```
jar -uf ear_file_name com.ibm.wmqfte.databaselogger.jee.ejb.jar
```

où *ear_file_name* est `com.ibm.wmqfte.databaselogger.jee.ear` ou `com.ibm.wmqfte.databaselogger.jee.oracle.ear` en fonction de votre base de données.

4. Pour déployer le fichier EAR sur le serveur d'applications, procédez comme suit à partir de la console d'administration WebSphere Application Server Community Edition .

- a) Sélectionnez **Applications > Deploy New** dans le menu **Console Navigation** .

- b) Dans la zone **Archive** , spécifiez le fichier EAR: `com.ibm.wmqfte.databaselogger.jee.ear` ou `com.ibm.wmqfte.databaselogger.jee.oracle.ear` en fonction de votre base de données.
- c) Laissez la zone **Plan** vide.
- d) Vérifiez que la case **Démarrer l'application après l'installation** est cochée.
- e) Cliquez sur **Install**. L'application de consigne de base de données JEE est installée et démarrée.

Configuration de l'accès utilisateur pour le consigne de base de données Java EE pour MFT

Lorsque vous configurez le consigne 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 consigne 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 consigne de base de données Java EE , 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 consigne 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 consigne utilise le mode Liaisons pour se connecter à IBM MQ et une connexion native pour se connecter à la base de données.

Consigne 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 consigne 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 consigne utilise le mode Liaisons pour se connecter à IBM MQ et une connexion client pour accéder à la base de données.

Le consigne 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 consigne 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 `fte1log`, 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 consigne 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 UNIX, 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 consigneur, 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 UNIX)
- 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 d'un consigneur de base de données Java EE

Pour migrer un consigneur de base de données Java EE sur WebSphere Application Server 7.0 à partir de IBM WebSphere MQ File Transfer Edition 7.0 vers IBM WebSphere MQ 7.5 ou ultérieure, procédez comme suit :

Procédure

1. Ouvrez la console WebSphere Application Server.
2. Cliquez sur **Applications > Types d'application > Applications d'entreprise**. Localisez l'application du consigneur de base de données IBM WebSphere MQ File Transfer Edition dans la liste des applications. Si cette application n'est pas déjà arrêtée, sélectionnez-la et cliquez sur **Arrêter**.
3. Prenez note des paramètres de configuration que vous avez définis précédemment pour le consigneur de base de données JEE. Vous aurez besoin de ces informations ultérieurement à l'étape «7», à la page 145.
 - a) Si vous avez initialement apporté des modifications aux paramètres par défaut des modules EJB lors de l'installation du consigneur de base de données (voir l'étape 9 pour plus d'informations), cliquez sur **Applications d'entreprise > WebSphere MQ File Transfer Edition > Entrées d'environnement pour les modules EJB** et notez les paramètres dans le panneau.

- b) Cliquez sur **Applications d'entreprise > WebSphere MQ File Transfer Edition consignateur de base de données > Liaisons d'écoute du bean géré par message** et notez la spécification d'activation utilisée, le **nom JNDI de la ressource cible** et le **nom JNDI de la destination**.
 - c) Cliquez sur **Applications d'entreprise > WebSphere MQ File Transfer Edition > Références de ressource** et notez les détails de la fabrique de connexions de file d'attente de rejet.
 - d) Cliquez sur **Applications d'entreprise > WebSphere MQ File Transfer Edition > Références d'entrée d'environnement de ressource** et notez les détails de la file d'attente de rejet.
4. Désinstallez l'application de consignateur de base de données IBM WebSphere MQ File Transfer Edition en cliquant sur **Applications > Types d'application > Applications d'entreprise**. Sélectionnez l'application de consignateur de base de données et cliquez sur **Désinstaller**.
 5. Facultatif : si vous utilisez plusieurs installations pour migrer vers IBM WebSphere MQ 7.5 ou une version ultérieure et que le chemin d'accès à la bibliothèque native est différent, modifiez le chemin d'accès en cliquant sur **Ressources > Fournisseurs JMS > WebSphere MQ**
 Par exemple, si le chemin d'accès à la bibliothèque native était: C:\Program Files\IBM\WebSphere MQ\java\lib, remplacez le chemin d'accès par: C:\Program Files\IBM\New MQ Installation Location\java\lib
 6. Facultatif : si vous utilisez plusieurs installations pour migrer vers IBM WebSphere MQ 7.5 ou ultérieure, vous devez associer le gestionnaire de files d'attente à la nouvelle installation à l'aide de la commande `setmqm`.
 7. Réinstallez l'application de consignateur de base de données à l'aide des informations de la rubrique Installation du consignateur de base de données IBM MQ Managed File Transfer Java EE avec WebSphere Application Server 7.0 et des informations que vous avez enregistrées précédemment à l'étape «3», à la page 144.
 8. Démarrez le nouveau consignateur de base de données en cliquant sur **Applications > Types d'application > Applications d'entreprise**. Sélectionnez l'application de consignateur de base de données et cliquez sur **Démarrer**.
 9. Pour vérifier la migration, vérifiez la base de données pour vous assurer que les entrées sont en cours d'écriture.

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 Java EE . 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 JEE. 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 7.0.1 et que vous migrez vers 7.0.4, vous devez migrer votre schéma de 7.0.1 vers 7.0.2, puis de 7.0.2 vers 7.0.3, puis de 7.0.3 vers 7.0.4. 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 7.0.2 et 7.0.3 ou entre les schémas 7.0.3 et 7.0.4, vous devez créer un nouveau schéma de base de données et y copier vos données existantes. Pour plus d'informations, voir la section [«Migration des tables de base de données sous Db2 sous z/OS vers IBM MQ 8.0»](#), à la page 22.
 - 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 la section [«Installation du consignateur de base de données Java EE pour MFT»](#), à la page 130.
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.

Pourquoi et quand exécuter cette tâche

Pour configurer le pont Connect:Direct, procédez comme suit:

Procédure

1. [«Choisissez les systèmes d'exploitation pour l'agent de pont et le noeud Connect:Direct»](#), à la page 147.
2. [«Choisissez et configurez un noeud Connect:Direct»](#), à la page 147.
3. [«Création et configuration d'un agent de pont Connect:Direct»](#), à la page 147.
4. [«Configurez le fichier ConnectDirectNodeProperties.xml pour inclure des informations sur les noeuds Connect:Direct distants»](#), à la page 148.
5. [«Configuration d'une connexion sécurisée entre l'agent de pont Connect:Direct et le noeud Connect:Direct»](#), à la page 149.

Choisissez les systèmes d'exploitation pour l'agent de pont et le noeud 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 System x
- L'agent se trouve sous Linux for System x et le noeud sous UNIX
- 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 System x 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 d'informations sur les versions de système d'exploitation prises en charge pour le pont Connect:Direct, voir la page Web [IBM MQ System Requirements](#).

Procédure

1. Choisissez un système exécutant Windows ou Linux sur System x pour installer l'agent de pont Connect:Direct.
2. 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.

Choisissez et configurez un noeud Connect:Direct

Avant de commencer

Vous devez avoir installé un noeud Connect:Direct avant de suivre ces instructions.

Procédure

1. Choisissez un noeud Connect:Direct avec lequel l'agent Managed File Transfer doit communiquer.
2. 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.
 - a) 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**.

Création et configuration d'un agent de pont Connect:Direct

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.

Procédure

1. Créez un agent de pont Connect:Direct à l'aide de la commande **fteCreateCDAgent** .
 - a) 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.
 - b) 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.
 - c) Utilisez les informations du [«fteCreateCDAgent \(créer un agent de pont Connect:Direct\)»](#), à la page 413 pour déterminer si vous devez spécifier une valeur pour le paramètre **cdTmpDir** .
2. 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 la section [«Mappage des données d'identification pour Connect:Direct à l'aide du fichier ConnectDirectCredentials.xml»](#), à la page 149.
 - Ecrivez un exit utilisateur pour effectuer un mappage de données d'identification pour votre pont Connect:Direct . Pour plus d'informations, voir la section [«Mappage des données d'identification pour Connect:Direct à l'aide de classes d'exit»](#), à la page 151.

Configurez le fichier `ConnectDirectNodeProperties.xml` pour inclure des informations sur les noeuds Connect:Direct distants

Avant de commencer

Vous devez avoir créé un agent de pont Connect:Direct avant de suivre ces instructions.

Procédure

Editez 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 admises sont les suivantes :

- Windows -le noeud s'exécute sur Windows
- UNIX -le noeud s'exécute sous UNIX ou Linux
-  z/OS, zos, os/390ou 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 la section [«Format du fichier de propriétés du noeud Connect:Direct»](#), à la page 603.

Configuration d'une connexion sécurisée entre l'agent de pont Connect:Direct et le noeud Connect:Direct

Pourquoi et quand exécuter cette tâche

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

Configurez une connexion sécurisée. 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»](#), à la page 55.

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.

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 la section [«Format de fichier de données d'identification Connect:Direct»](#), à la page 599. 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 UNIX , 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 146

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», à la page 599

Le fichier `ConnectDirectCredentials.xml` du répertoire de configuration Managed File Transfer Agent définit les noms d'utilisateur et les données d'identification utilisés par l'agent Connect:Direct pour s'autoriser avec un noeud Connect:Direct .

«fteCreateCDAgent (créer un agent de pont Connect:Direct)», à la page 413

La commande `fteCreateCDAgent` crée un Managed File Transfer Agent et sa configuration associée à utiliser avec le 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 la section «[Interface CDCredentialExit.java](#)», à la page 908.

Configuration de SSL ou TLS entre l'agent de pont Connect:Direct et le noeud Connect:Direct

Configurez l'agent de pont Connect:Direct et le noeud Connect:Direct pour qu'ils se connectent via le protocole SSL en créant un magasin de clés et un magasin de clés de confiance et en définissant les propriétés dans le fichier de propriétés de l'agent de pont Connect:Direct .

Pourquoi et quand exécuter cette tâche

Ces étapes incluent des instructions permettant d'obtenir vos clés signées par une autorité de certification. Si vous n'utilisez pas d'autorité de certification, vous pouvez générer un certificat autosigné. Pour plus d'informations sur la génération d'un certificat autosigné, voir [Utilisation de SSL ou TLS sur les systèmes UNIX et Windows](#).

Ces étapes incluent des instructions pour la création d'un magasin de clés et d'un magasin de clés de confiance pour l'agent de pont Connect:Direct . Si l'agent de pont Connect:Direct possède déjà un magasin de clés et un magasin de clés de confiance qu'il utilise pour se connecter de manière sécurisée aux gestionnaires de files d'attente IBM MQ , vous pouvez utiliser le magasin de clés et le magasin de clés de confiance existants lors de la connexion sécurisée au noeud Connect:Direct . Pour plus d'informations, voir la section «[Configuration du chiffrement SSL ou TLS pour MFT](#)», à la page 52.

Procédure

Pour le noeud Connect:Direct , procédez comme suit:

1. Générez une clé et un certificat signé pour le noeud Connect:Direct .
Pour ce faire, utilisez l'outil IBM Key Management fourni avec IBM MQ. Pour plus d'informations, voir [Utilisation avec SSL ou TLS](#).
2. Envoyez une demande à une autorité de certification pour que la clé soit signée. Vous recevez un certificat en retour.
3. Créez un fichier texte, par exemple `/test/ssl/certs/CACert`, qui contient la clé publique de votre autorité de certification.
4. Installez l'option Secure + sur le noeud Connect:Direct .
Si le noeud existe déjà, vous pouvez installer l'option Secure + en exécutant à nouveau le programme d'installation, en indiquant l'emplacement de l'installation existante et en choisissant d'installer uniquement l'option Secure +.
5. Créez un nouveau fichier texte ; par exemple, `/test/ssl/cd/keyCertFile/node_name.txt`.
6. Copiez le certificat que vous avez reçu de votre autorité de certification et la clé privée, qui se trouve dans `/test/ssl/cd/privateKeys/node_name.key`, dans le fichier texte.

Le contenu de `/test/ssl/cd/keyCertFile/node_name.txt` doit être au format suivant:

```
-----BEGIN CERTIFICATE-----
MIICnzCCAgigAwIBAgIBGjANBgkqhkiG9w0BAQUFADBeMQswCQYDVQQGEwJH0jES
MBAGA1UECBMJSgFtchNoaXJlMRAwDgYDVQQHEwdIdXJzbGV5M0wwCgYDVQQKEwNJ
Qk0xDjAMBgNVBAsTBURSVBUMQswCQYDVQQDEwJDTAeFw0xMTAzMDExNjIwNDZa
Fw0yMTAyMjYxNjIwNDZaMFAXCzAxBGNVBAYTAkdCMRIwEAYDVQQIEwIiYW1wc2hp
cmUxDDAKBgNVBAoTA01CTTEOMAwGA1UECzMFTVFGVEUxDzANBgNVBAMTBmJpbmJh
ZzCBnzANBgkqhkiG9w0BAQEFAA0BjQAwYkCgYEAvgP1QIk1U9ypSKD1Xo0Do1yk
EyMFXB0UpZrRiDVxjoSEC0vtWncJ199e+Vc4UpNybdyBu+Nkd1MNoFX4QxeQcLAFj
WnhakqCiQ+JIAD5AurhnriwChe0MV3kjA84GKH/1OSVqt1984mu/1DyS819XcfSSn
c00MsK1KbneVSCIV2XECawEAaAa7MHkwcQYDVR0TBAlwADAsBg1ghkgBhvhCAQ0E
HxYdT3Blb1NTTCBHZW51cmF0ZWQgQ2VydGlmawNhdGUwHQYDVR00BBYEFNXXMIpSc
csBXUniW4A3UirZnCRsv3MB8GA1UdIwQYMBaAFDXY8rmj41Vz5+FVAoQb++cns+B4
MA0GCSqGSIb3DQEBBQUAA4GBAFc7k1Xa4pGKYgwxhKpE3ZF6FNwy4vBXS216/ja
8h/v18+iv010Cl8t0ZOKSU95fyzLz0PKnCH7v+ItFSE3CIIEk9D1z2U6W091ICwn
```

```

17PL72Tdfal3kabwHYVf17IVcuL+VZsZ3HjLggP2qH09ZuJPspeT9+AxFVMLiaAb
8eHw
-----END CERTIFICATE-----
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,64A02DA15B6B6EF9

57kqxL0J/gRU0IQ6hVK2YN13B4E1jAi1gSme0I5ZpEIG8CHXISKB7/0cke2FTqsV
lvI99QyCxsDwoMnt5fj51v7aPmVeS60b0m+UlGre8B/Ze18JVj204K2Uh72rDCXE
5e6eFxsDUM207sQDy20euBVELJtM2k0kL1R0doQOS1U3XQNgJw/t3ZIx5hPXWEQT
rjRQ064BEhb+PzzxPF8uwzZ9IrkUK9BJ/UUnqC60dBR87IeA4pnJD1Jvb2ML7EN9Z
5Y+50hTKI80GvBvWX04fHyvIX5aslwhBoArXIS1AtNTprtPvoaP1zyIAeZ60Cvo/
SFo+A2UhmTEJe0JaZG2XZ3H495fAw/EHmjehzIACwukQ9nSIETgu4A1+CV64RJED
aYBCM8UjaAkbZDH5gn7+eBov0ssXAXWdyJBVhU0jXjvAj/e1h+kcSF1hax5D//AI
66nRMZzboSxNqkjcVd8wfdWp+bEjDzUaaarJTS71IFeLw7eJ8MNAkMGicDkycL0
EPBU9X5QnHKLK0fYHN/1WgUk8qt3UytFXXfzTXGF3EbsWbBupkT5e5+1YcX80VZ6
sHFPN1H1ucNy/riUcBy9iviVeodX8Iom0chSy05DK18bwZNjYtUP+CtYHNFU5BaD
I+1uU0AeJ+wjQYKT1WaeIGZ3VxuNITJu18y5qDTXXfX7vxM50oWxa6U5+AYuGUMg
/itPZmUmNrhjT7ghT6i1IQ0aBowXXKJB1Mmq/6BQXN2IhkD9ys2qrvM1hdi5nAf
egmdiG50loLnBRqWbFR+DykpAhK4SaDi2F52Uxovw3Lhiw8dQP71zQ==
-----END RSA PRIVATE KEY-----

```

7. Démarrez l'outil d'administration Secure +.

- Sur les systèmes Linux ou UNIX , exécutez la commande **spadmin.sh**.
- Sur les systèmes Windows , cliquez sur **Démarrer > Programmes > Sterling Commerce Connect:Direct > CD Secure + Admin Tool**

L'outil d'administration CD Secure + démarre.

8. Dans l'outil d'administration CD Secure +, cliquez deux fois sur **.Ligne** locale pour éditer les paramètres SSL ou TLS principaux.

- a) Sélectionnez **Activer le protocole SSL** ou **Activer le protocole TLS**, selon le protocole que vous utilisez.
- b) Sélectionnez **Désactiver la substitution**.
- c) Sélectionnez au moins une suite de chiffrement.
- d) Si vous souhaitez une authentification bidirectionnelle, remplacez la valeur de **Activer l'authentification client** par Yes.
- e) Dans la zone **Certificat racine accrédité** , entrez le chemin d'accès au fichier de certificat public de votre autorité de certification, /test/ssl/certs/CAcert.
- f) Dans la zone **Fichier de certificat de clé** , entrez le chemin d'accès au fichier que vous avez créé, /test/ssl/cd/keyCertFile/node_name.txt.

9. Cliquez deux fois sur le **.Ligne** du client pour éditer les paramètres SSL ou TLS principaux.

- a) Sélectionnez **Activer le protocole SSL** ou **Activer le protocole TLS**, selon le protocole que vous utilisez.
- b) Sélectionnez **Désactiver la substitution**.

Pour l'agent de pont Connect:Direct , procédez comme suit:

10. Créez un magasin de clés de confiance. Pour ce faire, vous pouvez créer une clé factice, puis la supprimer.

Vous pouvez utiliser les commandes suivantes:

```
keytool -genkey -alias dummy -keystore /test/ssl/fte/stores/truststore.jks
```

```
keytool -delete -alias dummy -keystore /test/ssl/fte/stores/truststore.jks
```

11. Importez le certificat public de l'autorité de certification dans le magasin de clés de confiance.

Vous pouvez utiliser la commande suivante :

```
keytool -import -trustcacerts -alias myCA
```

```
-file /test/ssl/certs/CAcert  
-keystore /test/ssl/fte/stores/truststore.jks
```

12. Editez le fichier de propriétés de l'agent de pont Connect:Direct .

Incluez les lignes suivantes n'importe où dans le fichier:

```
cdNodeProtocol=protocol  
cdNodeTruststore=/test/ssl/fte/stores/truststore.jks  
cdNodeTruststorePassword=password
```

Dans l'exemple de cette étape, *protocol* est le protocole que vous utilisez, SSL ou TLS, et *password* est le mot de passe que vous avez spécifié lors de la création du magasin de clés de confiance.

13. Si vous souhaitez une authentification bidirectionnelle, créez une clé et un certificat pour l'agent de pont Connect:Direct .

a) Créez un magasin de clés et une clé.

Vous pouvez utiliser la commande suivante :

```
keytool -genkey -keyalg RSA -alias agent_name  
-keystore /test/ssl/fte/stores/keystore.jks  
-storepass password -validity 365
```

b) Générez une demande de signature.

Vous pouvez utiliser la commande suivante :

```
keytool -certreq -v -alias agent_name  
-keystore /test/ssl/fte/stores/keystore.jks -storepass password  
-file /test/ssl/fte/requests/agent_name.request
```

c) Importez le certificat que vous avez reçu à l'étape précédente dans le magasin de clés. Le certificat doit être au format x.509 .

Vous pouvez utiliser la commande suivante :

```
keytool -import -keystore /test/ssl/fte/stores/keystore.jks  
-storepass password -file certificate_file_path
```

d) Editez le fichier de propriétés de l'agent de pont Connect:Direct .

Incluez les lignes suivantes n'importe où dans le fichier:

```
cdNodeKeystore=/test/ssl/fte/stores/keystore.jks  
cdNodeKeystorePassword=password
```

Dans l'exemple de cette étape, *password* est le mot de passe que vous avez spécifié lors de la création du magasin de clés.

Tâches associées

«Configuration du pont Connect:Direct», à la page 146

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 .

Spécification du processus Connect:Direct à démarrer à l'aide du fichier ConnectDirectProcessDefinition.xml

Indiquez le processus Connect:Direct à démarrer dans le cadre d'un transfert Managed File Transfer . Managed File Transfer fournit un fichier XML que vous pouvez éditer pour spécifier des définitions de processus.

Pourquoi et quand exécuter cette tâche

La commande **fteCreateCDAgent** crée le fichier `ConnectDirectProcessDefinitions.xml` dans le répertoire de configuration de l'agent `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent_name`. Avant de pouvoir appeler des processus `Connect:Direct` définis par l'utilisateur à partir de l'agent de pont `Connect:Direct`, vous devez configurer des définitions de processus en éditant ce fichier.

Pour chaque processus à appeler dans le cadre d'un transfert via le pont `Connect:Direct`, procédez comme suit:

Procédure

1. Définissez le processus `Connect:Direct` que l'agent de pont `Connect:Direct` doit appeler dans le cadre du transfert et sauvegardez le modèle de processus dans un fichier.
2. Ouvrez le fichier `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent_name/ConnectDirectProcessDefinitions.xml` dans n'importe quel éditeur de texte.
3. Créez un élément `<processSet>`.
4. Dans l'élément `<processSet>`, créez un élément `<condition>`.
5. Dans l'élément `<condition>`, créez un ou plusieurs éléments qui définissent une condition à laquelle la demande de transfert doit correspondre pour appeler le processus `Connect:Direct` que vous avez défini à l'étape 1. Ces éléments peuvent être des éléments `<match>` ou des éléments `<defined>`.
 - Utilisez un élément `<match>` pour indiquer que la valeur d'une variable doit correspondre à un modèle. Créez l'élément `<match>` avec les attributs suivants:
 - `variable` -nom de la variable dont la valeur est comparée. La variable est un symbole intrinsèque. Pour plus d'informations, voir la section «[Variables de substitution à utiliser avec les processus Connect:Direct définis par l'utilisateur](#)», à la page 680.
 - `value` -modèle à comparer à la valeur de la variable spécifiée.
 - Facultatif: `pattern` -type de canevas utilisé par la valeur de l'attribut `value`. Ce type de canevas peut être `wildcard` ou `regex`. Cet attribut est facultatif et la valeur par défaut est `wildcard`.
 - Utilisez un élément `<defined>` pour indiquer qu'une variable doit avoir une valeur définie. Créez l'élément `<defined>` avec l'attribut suivant:
 - `variable` -nom de la variable pour laquelle une valeur doit être définie. La variable est un symbole intrinsèque. Pour plus d'informations, voir la section «[Variables de substitution à utiliser avec les processus Connect:Direct définis par l'utilisateur](#)», à la page 680.
6. Dans l'élément `<processSet>`, créez un élément `<process>`.
7. Dans l'élément `<process>`, créez un élément `<transfer>`.

L'élément `transfer` spécifie le processus `Connect:Direct` que l'agent de pont `Connect:Direct` appelle dans le cadre du transfert. Créez l'élément `<transfer>` avec l'attribut suivant:

 - `process` - emplacement du processus `Connect:Direct` que vous avez défini à l'étape 1. L'emplacement de ce fichier est spécifié avec un chemin d'accès absolu ou relatif au répertoire `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent_name`.

Résultats

Lors de la recherche d'une correspondance de condition, l'agent de pont Connect:Direct effectue une recherche 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 146

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 des définitions de processus Connect:Direct», à la page 605

Le fichier ConnectDirectProcessDefinitions.xml du répertoire de configuration de l'agent de pont Connect:Direct spécifie le processus Connect:Direct défini par l'utilisateur à démarrer dans le cadre du transfert de fichiers.

«fteCreateCDAgent (créer un agent de pont Connect:Direct)», à la page 413

La commande fteCreateCDAgent crée un Managed File Transfer Agent et sa configuration associée à utiliser avec le pont Connect:Direct .

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 156 et l'étape «3», à la page 156 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 156.
 - e) Sur la machine principale, ajoutez l'agent en tant que 'Service générique' sous contrôle MSCS.
Pour ce faire :

- 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 [Insertion d'un gestionnaire de files d'attente sous contrôle MSCS](#).
 - 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 157 et l'étape «3», à la page 157 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 157.
 - f) Ajoutez l'agent en tant que service générique sous le contrôle de MSCS.
Pour ce faire :
 - 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.

Administration d'MFT

Utilisez les commandes Managed File Transfer pour administrer Managed File Transfer. Vous pouvez également utiliser le IBM MQ Explorer pour certaines des tâches d'administration.

Démarrer le transfert en plaçant un message dans une file d'attente de commandes d'agent

Vous pouvez également démarrer un transfert de fichier en plaçant un message de transfert de fichier dans la file d'attente de commandes de l'agent source. Exemple de nom de file d'attente de commandes: SYSTEM.FTE.COMMAND.AGENT01. Vous devez vous assurer que le message atteint la file d'attente de commandes de l'agent source correct ; si le message est reçu par un agent qui ne correspond pas aux informations source dans le XML, le message est rejeté.

Le XML de la demande de transfert doit être conforme au schéma `FileTransfer.xsd` et utiliser l'élément `<request>` comme élément racine. Pour plus d'informations sur la structure et le contenu d'un message de demande de transfert, voir [Format de message de demande de transfert de fichier](#). La façon dont vous placez le message de demande de transfert dans une file d'attente de commandes d'agent est spécifique à la tâche. Par exemple, vous pouvez utiliser l'API IBM MQ Java pour placer un message dans la file d'attente à l'aide d'un programme.

Démarrage d'un agent MFT

Avant de pouvoir utiliser un agent Managed File Transfer pour un transfert de fichier, vous devez d'abord démarrer l'agent.

Pourquoi et quand exécuter cette tâche

Vous pouvez démarrer un Managed File Transfer Agent à partir de la ligne de commande. Dans ce cas, le processus de l'agent s'arrête lorsque vous vous déconnectez du système.

ULW Sous UNIX, Linux, and Windows, vous pouvez configurer un agent pour qu'il continue de s'exécuter lorsque vous vous déconnectez du système et qu'il puisse continuer à recevoir des transferts de fichiers.

z/OS Sous z/OS, vous pouvez configurer l'agent pour qu'il démarre en tant que tâche démarrée à partir de JCL sans qu'une session interactive soit nécessaire.

Notez que si un agent rencontre une erreur irrémédiable lors de son exécution, une capture de données de première défaillance (FDC) est générée et l'agent est arrêté.

Procédure

- Pour démarrer un agent à partir de la ligne de commande, utilisez la commande **fteStartAgent**. Pour plus d'informations, voir la section [«fteStartAgent»](#), à la page 540.
- **ULW**
 - Pour configurer un agent afin qu'il continue de s'exécuter lorsque vous vous déconnectez du système:
 - **Windows** Sous Windows, configurez l'agent pour qu'il s'exécute en tant que service Windows. Pour plus d'informations, voir la section [«Démarrage d'un agent MFT en tant que service Windows»](#), à la page 159.
 - **Linux** **UNIX** Sous UNIX et Linux, configurez l'agent pour qu'il démarre automatiquement lors d'un réamorçage à l'aide d'un fichier script. Pour plus d'informations, voir la section [«Démarrage d'un agent MFT au démarrage du système UNIX et Linux»](#), à la page 160.
- **z/OS**
 - Sous z/OS, configurez l'agent pour qu'il démarre en tant que tâche démarrée à partir de JCL sans qu'une session interactive soit nécessaire. Pour plus d'informations, voir la section [«Démarrage d'un agent MFT sous z/OS»](#), à la page 162.

Vous pouvez démarrer un agent en tant que service Windows de sorte que lorsque vous vous déconnectez de Windows, votre agent continue de s'exécuter et puisse recevoir des transferts de fichiers.

Pourquoi et quand exécuter cette tâche

Sous Windows, lorsque vous démarrez un agent à partir de la ligne de commande, le processus de l'agent s'exécute à l'aide du nom d'utilisateur que vous avez utilisé pour vous connecter à Windows. Lorsque vous vous déconnectez du système, le processus de l'agent s'arrête. Pour empêcher l'arrêt de l'agent, vous pouvez configurer un agent pour qu'il s'exécute en tant que service Windows. L'exécution en tant que service Windows vous permet également de configurer les agents pour qu'ils soient démarrés automatiquement lorsque l'environnement Windows démarre ou est redémarré.

Procédez comme suit pour démarrer un agent qui s'exécute en tant que service Windows. Vous devez exécuter Managed File Transfer sur l'une des versions Windows prises en charge pour exécuter l'agent en tant que service Windows. Pour la liste des environnements pris en charge, voir [IBM MQ System Requirements](#).

Les étapes exactes varient selon que vous avez déjà créé un agent ou que vous créez un agent. Les deux options sont décrites dans les étapes suivantes.

Procédure

1. Si vous créez un agent Managed File Transfer, utilisez la commande **fteCreateAgent**, **fteCreateCDAgent** ou **fteCreateBridgeAgent**. Indiquez le paramètre **-s** pour exécuter l'agent en tant que service Windows. Dans l'exemple suivant, l'agent AGENT1 est créé avec un gestionnaire de files d'attente d'agent QMGR1. Le service Windows s'exécute à l'aide du nom d'utilisateur fteuser, qui est associé à un mot de passe ftepassword.

```
fteCreateAgent -agentName AGENT1 -agentQMGR QMGR1 -s -su fteuser -sp ftepassword
```

Vous pouvez éventuellement spécifier un nom pour le service après le paramètre **-s**. Si vous n'indiquez pas de nom, le service est nommé mqmftAgentAGENTQMGR, où AGENT est le nom d'agent que vous avez indiqué et QMGR est le nom de votre gestionnaire de files d'attente d'agent. Dans cet exemple, le nom par défaut du service est mqmftAgentAGENT1QMGR1.

Remarque : Le compte utilisateur Windows que vous spécifiez à l'aide du paramètre **-su** doit disposer des droits **Log on as a service**. Pour plus d'informations sur la configuration, voir «[Conseils pour l'exécution d'un agent ou d'un consignateur MFT en tant que service Windows](#)», à la page 333.

Pour plus d'informations, voir «[Agent fteCreate \(créer un agent MFT\)](#)», à la page 399, «[fteCreateCDAgent \(créer un agent de pont Connect:Direct\)](#)», à la page 413 ou «[fteCreateBridgeAgent \(créer et configurer un agent de pont de protocole MFT\)](#)», à la page 403.

2. Si vous avez suivi l'étape précédente pour créer un agent, exécutez les commandes MQSC générées par la commande **fteCreateAgent**, **fteCreateCDAgent** ou **fteCreateBridgeAgent**. Ces commandes créent les files d'attente IBM MQ requises par l'agent.
Par exemple, pour un agent nommé AGENT1, un gestionnaire de files d'attente d'agent nommé QMGR1 et un gestionnaire de files d'attente de coordination nommé COORDQMGR1, exécutez la commande suivante:

```
runmqsc QMGR1 MQ_DATA_PATH\mqft\config\COORDQMGR1\agents\AGENT1\AGENT1_create.mqsc
```

3. Si vous n'avez pas suivi les étapes précédentes pour créer un agent et que vous souhaitez à la place configurer un agent existant pour qu'il s'exécute en tant que service Windows, arrêtez d'abord votre agent s'il est en cours d'exécution, puis modifiez sa configuration.
 - a) L'exemple suivant utilise un agent nommé AGENT1. Exécutez ensuite la commande suivante :

```
fteStopAgent AGENT1
```

- b) Utilisez la commande **fteModifyAgent** pour configurer l'agent pour qu'il s'exécute en tant que service Windows :

```
fteModifyAgent -agentName AGENT1 -s -su fteuser -sp ftepassword
```

Pour plus d'informations, voir la section [«Agent fteModify\(exécutez un agent MFT en tant que service Windows \)»](#), à la page 502.

4. Démarrez votre agent à l'aide de la commande **fteStartAgent** . Vous pouvez également utiliser l'outil Windows Services, disponible à partir des outils d'administration du panneau de configuration, sélectionné dans le menu Démarrer du bureau Windows , pour démarrer le service.

```
fteStartAgent AGENT1
```

Le service continue de s'exécuter même si vous vous déconnectez de Windows. Pour vous assurer que le service redémarre également lorsque Windows redémarre après un arrêt, la zone **Type de démarrage** de l'outil Windows Services est définie sur **Automatique** par défaut. Remplacez cette valeur par **Manuel** si vous ne souhaitez pas que le service redémarre lorsque Windows redémarre.

5. Facultatif : Pour arrêter l'agent, utilisez la commande [fteStopAgent](#) ou l'outil Windows Services. Par exemple, à partir de la ligne de commande, exécutez la commande suivante:

```
fteStopAgent AGENT1
```

- Lorsque vous exécutez la commande **fteStopAgent** en tant que service, la commande s'exécute toujours à l'aide du paramètre **-i** , que vous ayez ou non spécifié ce paramètre. Le paramètre **-i** arrête immédiatement l'agent sans effectuer les transferts en cours. Cela est dû à une limitation du service Windows .

Que faire ensuite

Si vous rencontrez des problèmes lors du démarrage du service Windows , voir [«Conseils pour l'exécution d'un agent ou d'un consignateur MFT en tant que service Windows»](#), à la page 333. Cette rubrique décrit également l'emplacement des fichiers journaux du service Windows .

Linux

UNIX

Démarrage d'un agent MFT au démarrage du système

UNIX et Linux

Un Managed File Transfer Agent peut être configuré pour démarrer au démarrage du système sur UNIX. Lorsque vous déconnectez un système UNIX , votre agent continue de s'exécuter et peut recevoir des transferts de fichiers.

Une fois que vous avez créé et configuré un agent à l'aide de l'une des commandes Managed File Transfer suivantes: **fteCreateAgent**, **fteCreateCDAgent**, ou **fteCreateBridgeAgent**, vous pouvez le configurer pour qu'il démarre automatiquement lors d'un réamorçage sur les machines UNIX à l'aide d'un fichier script qui s'exécute simplement:

```
su -l mqmft_user -c mq_install_root/bin/fteStartAgent agent_name
```

Où *mq_install_root* est le répertoire racine de l'installation Managed File Transfer requise, la valeur par défaut est: /opt/mqm et *nom_agent* est le nom du Managed File Transfer Agent à démarrer. L'utilisation de ce fichier script varie en fonction du système d'exploitation UNIX spécifique.

Solaris

Solaris

Sous Solaris, procédez comme suit:

1. Exécutez la commande suivante et gardez une trace du chemin renvoyé:

```
which sh
```

Par exemple, le chemin peut être `/usr/bin/sh`

2. En tant qu'ID utilisateur `root` du système, créez votre propre fichier `/etc/init.d/startmqmft`.
3. Editez ce fichier et ajoutez-y les lignes de script, en utilisant le chemin renvoyé à l'étape 1 comme première ligne du script:

```
#!/usr/bin/sh
su mqmft_user mq_install_root/bin/fteStartAgent agent_name
```

4. Rendez le fichier exécutable, par exemple:

```
chmod 755 /etc/init.d/startmqmft
```

5. Créez un lien symbolique entre le fichier et le répertoire `rc3.d` :

```
ln -s /etc/init.d/startmqmft /etc/rc3.d/S98startmqmft
```

Le préfixe `S` indique l'état par défaut Démarré pour Solaris. `98` est un numéro de séquence. Le suffixe est le nom de fichier de `init.d`

HP-UX

HP-UX

Sous HP-UX, procédez comme suit:

1. Créez un fichier appelé `/sbin/init.d/mqmft` avec son contenu:

```
#!/bin/sh
su -l mqmft_user -c mq_install_root/bin/fteStartAgent agent_name
```

2. Créez un fichier appelé `/etc/rc.config.d/mqmft` avec son contenu:

```
MQMFT=1
```

3. Liez le fichier au répertoire `rc3.d` et démarrez l'agent:

```
ln -s /sbin/init.d/mqmft /sbin/rc3.d/S84mqmft
```

Linux

Linux

Pour les systèmes Linux, vous pouvez démarrer des applications de plusieurs manières au cours du processus d'amorçage du système. En général, tenez compte des étapes suivantes:

1. Créez un fichier appelé `/etc/rc.mqmft` avec son contenu:

```
#!/bin/sh
su -l mqmft_user -c mq_install_root/bin/fteStartAgent agent_name"
```

Où `mqmft_user` est l'ID utilisateur sous lequel le processus d'agent doit s'exécuter. Cet ID utilisateur doit être membre du groupe `mqm`.

2. Rendez le fichier exécutable, par exemple:

```
chmod 755 /etc/rc.mqmf
```

3. Ajoutez ensuite la ligne suivante à `/etc/inittab`:

```
mqmf:5:boot:/etc/rc.mqmf
```

Vous pouvez également démarrer un agent lors de l'amorçage sur Linux en ajoutant les lignes de script au fichier `/etc/rc.d/rc.local` ou sur Linux SuSe, en ajoutant les lignes de script au fichier `/etc/init.d/boot.local`. Vous devez sélectionner la méthode qui convient le mieux à votre environnement. Voici quelques informations supplémentaires sur les autres méthodes de démarrage d'un agent lors du démarrage sur des distributions Linux spécifiques prises en charge:

SLES 10 et 11

Pour les systèmes SUSE Linux Enterprise Server (SLES) 10 et 11, procédez comme suit:

1. En tant qu'ID utilisateur root du système, créez votre propre fichier `/etc/init.d/rc.rclocal`.
2. Ajoutez les lignes suivantes au fichier `rc.rclocal`:

```
#!/bin/sh
### BEGIN INIT INFO
# Provides: rc.rclocal
# Required-Start: $network $syslog
# Required-Stop: $network $syslog
# Default-Stop: 0 1 2 6
# Description: MQMFT agent startup
### END INIT INFO
su -l mqmf_user -c mq_install_root/bin/fteStartAgent agent_name"
```

3. Exécutez les commandes suivantes :

```
chmod 755 rc.rclocal
chkconfig --add rc.rclocal
```

Démarrage d'un agent MFT sous z/OS

Sous z/OS, en plus d'exécuter la commande **fteStartAgent** à partir d'une session UNIX System Services, vous pouvez démarrer un agent en tant que tâche démarrée à partir de JCL sans avoir besoin d'une session interactive.

Une tâche démarrée est utilisée car elle s'exécute sous un ID utilisateur spécifique et n'est pas affectée par la déconnexion des utilisateurs.

Remarque :   Les tâches démarrées sont généralement exécutées sous un administrateur qui ne dispose peut-être pas de privilèges de connexion et il n'est donc pas possible de se connecter au système z/OS en tant qu'utilisateur sous lequel l'agent est exécuté. Les commandes **fteStartAgent**, **fteStopAgent** et **fteSetAgentTraceLevel**, ainsi que la commande **fteShowAgentDetails** avec le paramètre **-d** spécifié, ne peuvent pas être émises pour cet agent.

Depuis IBM MQ 9.0.2 et IBM MQ 9.0.0 Fix Pack 1, la propriété d'agent **adminGroup** peut être utilisée avec les agents Managed File Transfer sous z/OS. Vous pouvez définir un groupe de gestionnaires de sécurité, par exemple MFTADMIN, puis ajouter l'ID utilisateur de la tâche démarrée et les ID TSO d'administrateur à ce groupe. Editez le fichier de propriétés de l'agent et définissez la propriété **adminGroup** pour qu'elle soit le nom de ce groupe de gestionnaires de sécurité.

```
adminGroup=MFTADMIN
```

Les membres de ce groupe peuvent ensuite émettre les commandes **fteStartAgent**, **fteStopAgent** et **fteSetAgentTraceLevel**, ainsi que la commande **fteShowAgentDetails** avec le paramètre **-d** spécifié, pour l'agent qui s'exécute en tant que tâche démarrée.

Pour plus d'informations, voir la propriété **adminGroup** dans «Le fichier MFT agent.properties», à la page 564.

En tant qu'application Java, un agent est une application UNIX System Services que vous pouvez exécuter à partir du JCL à l'aide du membre BFGAGSTP, à partir d'un fichier de bibliothèque PDSE de commande Managed File Transfer généré pour un agent. Pour plus d'informations sur la création d'un fichier de bibliothèque PDSE de commande MFT et sa personnalisation pour l'agent requis, voir «Création d'un fichier de commandes d'agent ou de consignateur MFT», à la page 64.

Activer les connexions à distance d'agent avec IBM MQ Advanced for z/OS, Value Unit Edition uniquement

V 9.0.3 MQ Adv. VUE

Un agent MFT sous z/OS, exécuté sous l'identificateur de produit (PID) IBM MQ Advanced for z/OS VUE, peut se connecter à un gestionnaire de files d'attente sous z/OS à l'aide d'une connexion client.

Lorsqu'un agent démarre, il écrit un message BFGPR0137I dans son journal des événements (output0.log) indiquant le PID sous lequel il s'exécute. Voici un exemple de ce message:

```
BFGPR0137I: L'enregistrement des données d'utilisation du produit a démarré pour le produit'MQ z/OS MFT', ID produit'5655-MF9'.
```

Pour plus de détails sur les produits IBM MQ, les valeurs de PID qui leur sont associées et les classifications d'exportation, voir [Identificateurs de produit IBM MQ et informations d'exportation](#).

Un agent MFT sous z/OS, s'exécutant sous n'importe quel autre PID, peut se connecter à un gestionnaire de files d'attente local à l'aide d'une connexion de liaisons uniquement.

Un agent MFT sous z/OS ne peut se connecter qu'à un gestionnaire de files d'attente, également exécuté sous z/OS, quel que soit l'ID de produit MFT.

Si un agent IBM MQ Advanced for z/OS VUE tente de se connecter à un gestionnaire de files d'attente qui ne s'exécute pas sous z/OS, le message [BFGMQ1044E](#) suivant est émis et le démarrage de l'agent est arrêté:

```
BFGMQ1044E: La connexion du client d'agent sous z/OS doit être établie avec un gestionnaire de files d'attente sous z/OS
```

Référence associée

«Arrêt d'un agent MFT sous z/OS», à la page 218

Si vous exécutez un Managed File Transfer Agent sous z/OS en tant que tâche démarrée à partir de JCL, l'agent accepte les commandes d'opérateur z/OS **MODIFY** et **STOP**, en plus de la commande **fteStopAgent**.

«Le fichier MFT agent.properties», à la page 564

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.

Démarrage d'un nouveau transfert de fichiers

Vous pouvez démarrer un nouveau transfert de fichier à partir de IBM MQ Explorer ou de la ligne de commande et vous pouvez choisir de transférer un seul fichier ou plusieurs fichiers dans un groupe.

Pourquoi et quand exécuter cette tâche

Pour démarrer un nouveau transfert de fichier à partir de la ligne de commande, voir [fteCreateTransfer command](#).

Pour démarrer un nouveau transfert de fichier à l'aide de l'assistant **Créer un transfert de fichier géré** dans IBM MQ Explorer, procédez comme suit:

Procédure

1. Dans la vue Navigator , cliquez sur **Transfert de fichier géré. Managed File Transfer Central** s'affiche dans la vue Contenu.
2. Tous vos gestionnaires de files d'attente de coordination sont affichés dans la vue Navigator . Développez le nom du gestionnaire de file d'attente de coordination sur lequel l'agent que vous souhaitez utiliser pour le transfert est enregistré. Si vous êtes actuellement connecté à un gestionnaire de files d'attente de coordination autre que celui que vous souhaitez utiliser pour le transfert, cliquez avec le bouton droit de la souris sur le nom du gestionnaire de files d'attente de coordination dans la vue Navigator , puis cliquez sur **Déconnexion**. Cliquez ensuite avec le bouton droit de la souris sur le nom du gestionnaire de files d'attente de coordination à utiliser, puis cliquez sur **Connexion**.
3. Démarrez l'assistant **Créer un nouveau transfert de fichier géré** à l'aide de l'une des méthodes suivantes:
 - a) Cliquez avec le bouton droit de la souris sur le nom de l'un des noeuds suivants dans la vue Navigator : le gestionnaire de files d'attente de coordination approprié, **Modèles de transfert**, **Journal de transfert** ou **Transferts en attente**. Cliquez ensuite sur **Nouveau transfert** pour démarrer l'assistant.
 - b) Cliquez sur **Fichier > Nouveau > Autres > Assistants Managed File Transfer > Assistant Nouveau transfert**
4. Suivez les instructions des panneaux de l'assistant. Une aide contextuelle est également fournie pour chaque panneau. Pour accéder à l'aide contextuelle sur Windows, appuyez sur F1. Sous Linux, appuyez sur Ctrl+F1 ou sur Shift+F1.

Utilisation de fichiers de définition de transfert

Vous pouvez spécifier un fichier de définition de transfert qui peut être utilisé pour créer un transfert de fichier. Le fichier de définition de transfert est un fichier XML qui définit tout ou partie des informations requises pour créer le transfert.

Les fichiers de définition de transfert sont utiles lorsque vous souhaitez spécifier plusieurs fichiers source et plusieurs fichiers de destination dans une seule opération de transfert. Vous pouvez utiliser un fichier de définition de transfert pour soumettre un transfert de fichier complexe. Vous pouvez réutiliser et partager le fichier de définition de transfert.

Vous pouvez utiliser deux formats pour un fichier de définition de transfert et, bien que ces formats varient légèrement, ils sont tous deux conformes au schéma `FileTransfer.xsd` . Ce schéma se trouve dans le répertoire `samples\schema` de l'installation Managed File Transfer .

Les deux formats suivants de fichiers de définition de transfert sont pris en charge:

- Définition des fichiers source et cible pour un transfert. Cette définition utilise un élément `<transferSpecifications>` comme racine.
- Définition de l'ensemble du transfert, y compris les fichiers source et de destination et les agents source et de destination. Cette définition utilise un élément `<request>` comme racine.
 - Les fichiers de ce format peuvent être générés à partir de la commande **fteCreateTransfer** à l'aide du paramètre **-gt** .

L'exemple suivant illustre un format de fichier de définition de transfert qui spécifie uniquement les fichiers source et cible d'un transfert:

```
<?xml version="1.0" encoding="UTF-8"?>
<transferSpecifications xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <item checksumMethod="MD5" mode="text">
    <source recursive="false" disposition="leave">
      <file>textTransferTest.txt</file>
    </source>
```

```

    <destination type="directory" exist="overwrite">
      <file>c:\targetfiles</file>
    </destination>
  </item>
</transferSpecifications>

```

Pour soumettre ce format de fichier de définition de transfert, vous devez spécifier les agents source et cible sur la ligne de commande:

```

fteCreateTransfer -sa AGENT1 -sm agent1qm -da AGENT2 -dm agent2qm -td
c:\definitions\example1.xml

```

L'exemple suivant illustre un format de fichier de définition de transfert qui spécifie toutes les informations requises pour un transfert:

```

<?xml version="1.0" encoding="UTF-8"?>
<request version="3.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>example.com.</hostName>
      <userID>fteuser</userID>
    </originator>
    <sourceAgent agent="AGENT1" QMgr="agent1qm"/>
    <destinationAgent agent="AGENT2" QMgr="agent2qm"/>
    <transferSet>
      <item mode="binary" checksumMethod="MD5">
        <source recursive="false" disposition="leave">
          <file>c:\sourcefiles\*.jpg</file>
        </source>
        <destination type="directory" exist="error">
          <file>/targetfiles/images</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>

```

Vous pouvez générer un fichier avec ce format à l'aide du paramètre **-gt** de la commande **fteCreateTransfer**. Lorsque vous soumettez un fichier de définition de transfert avec ce format, vous n'avez pas besoin de spécifier autre chose sur la ligne de commande:

```

fteCreateTransfer -td c:\definitions\example2.xml

```

Vous pouvez remplacer les informations d'agent source et de destination sur la ligne de commande en transmettant les paramètres normaux en plus du fichier de définition de transfert. Exemple :

```

fteCreateTransfer -da AGENT9 -dm agent9qm -td c:\definitions\example2.xml

```

Cet exemple utilise les options de ligne de commande pour remplacer l'agent de destination défini dans le fichier de définition de transfert par **AGENT9** et le gestionnaire de files d'attente de destination défini dans le fichier de définition de transfert sous la forme **agent9qm**.

Les deux formats décrits peuvent contenir un ou plusieurs éléments < item>. Pour plus d'informations sur l'élément < item>, voir [Format de message de demande de transfert de fichier](#). Chacun de ces éléments de transfert définit une paire de fichiers source et de destination avec des attributs supplémentaires pour contrôler le comportement du transfert. Par exemple, vous pouvez spécifier le comportement suivant:

- Indique si le transfert utilise un total de contrôle
- Indique si le transfert est de type texte ou binaire
- Indique si le fichier source doit être supprimé une fois le transfert terminé
- Indique s'il faut remplacer le fichier de destination s'il existe

L'avantage de l'utilisation des fichiers de définition de transfert est que vous pouvez spécifier des options supplémentaires qui ne sont pas disponibles à partir de la ligne de commande. Par exemple, lorsque

vous effectuez des transferts de message à fichier, vous pouvez spécifier l'attribut `groupId` à l'aide d'un fichier de définition de transfert. Cet attribut indique l'ID groupe IBM MQ des messages lus à partir de la file d'attente. Un autre avantage des fichiers de définition de transfert est que vous pouvez spécifier des options différentes pour chaque paire de fichiers. Par exemple, vous pouvez indiquer si un total de contrôle est utilisé ou si le fichier est transféré en mode texte ou binaire, fichier par fichier. Si vous utilisez la ligne de commande, les mêmes options s'appliquent à chaque fichier d'un transfert.

Exemple :

```
<item checksumMethod="none" mode="binary">
  <source disposition="leave">
    <file>c:\sourcefiles\source1.doc</file>
  </source>
  <destination type="file" exist="error">
    <file>c:\destinationfiles\destination1.doc</file>
  </destination>
</item>

<item checksumMethod="MD5" mode="text">
  <source disposition="delete">
    <file>c:\sourcefiles\source2.txt</file>
  </source>
  <destination type="file" exist="overwrite">
    <file encoding="UTF8" EOL="CRLF">c:\destinationfiles\destination2.txt</file>
  </destination>
</item>

<item checksumMethod="none" mode="text">
  <source recursive="false" disposition="leave">
    <file>c:\originfiles\source3.txt</file>
  </source>
  <destination type="file" exist="overwrite">
    <file>c:\targetfiles\destination3.txt</file>
  </destination>
</item>
```

z/OS Vous pouvez utiliser des éléments pour transférer un fichier d'un système réparti vers un système z/OS :

```
z/OS
<item checksumMethod="none" mode="text">
  <source recursive="false" disposition="leave">
    <file>textTransferTest.txt</file>
  </source>
  <destination type="dataset" exist="overwrite">
    <file encoding="IBM-1047">//TEXT.TRANS.TEST</file>
  </destination>
</item>
```

z/OS Cet exemple transfère le fichier `textTransferTest.txt` de l'agent source vers l'ensemble de données `//TEXT.TRANS.TEST` sur l'agent cible en mode texte. Ce transfert convertit les données source du codage par défaut de l'agent source (aucun attribut de codage source n'est spécifié) en page de codes: IBM-1047.

Création d'un transfert de fichiers planifié

Vous pouvez planifier un nouveau transfert de fichiers à partir de l'explorateur IBM MQ Explorer ou de la ligne de commande. Le transfert planifié peut contenir un seul fichier ou plusieurs fichiers dans un groupe. Vous pouvez effectuer un transfert de fichier planifié une seule fois ou répéter le transfert plusieurs fois.

Pourquoi et quand exécuter cette tâche

Vous pouvez configurer une planification de transfert de fichiers pour qu'elle se produise une fois ou aux intervalles suivants:

- Toutes les minutes

- Horaire
- Quotidien
- Toutes les semaines
- Mensuel
- Annuel

Vous pouvez ensuite spécifier les occurrences à arrêter aux points suivants:

- A une heure et une date définies
- Après un nombre défini d'occurrences

Vous pouvez également indiquer que les occurrences se poursuivent indéfiniment.

V 9.0.0.9 Si un transfert planifié s'exécute à la même heure chaque jour, utilisez l'attribut **adjustScheduleTimeForDaylightSaving** dans le fichier de propriétés de l'agent pour ajuster l'heure à laquelle la planification a lieu lorsque les horloges changent. Pour plus d'informations, voir [Le fichier MFT agent.properties](#).

Pour créer un transfert de fichier planifié à l'aide de la ligne de commande, utilisez les paramètres de planification (**-tb, -ss, -oi, -of, -ocet -es**) pour la [commande de transfertfiteCreate](#).

Pour créer un transfert de fichier planifié à l'aide de l'assistant **Créer un transfert de fichier géré** dans IBM MQ Explorer, procédez comme suit:

Procédure

1. Dans la vue Navigator , cliquez sur **Transfert de fichier géré. Managed File Transfer Central** s'affiche dans la vue Contenu.
2. Tous vos gestionnaires de files d'attente de coordination sont affichés dans la vue Navigator . Développez le nom du gestionnaire de file d'attente de coordination sur lequel l'agent que vous souhaitez utiliser pour le transfert est enregistré. Si vous êtes actuellement connecté à un gestionnaire de files d'attente de coordination autre que celui que vous souhaitez utiliser pour le transfert, cliquez avec le bouton droit de la souris sur le nom du gestionnaire de files d'attente de coordination dans la vue Navigator , puis cliquez sur **Déconnexion**. Cliquez ensuite avec le bouton droit de la souris sur le nom du gestionnaire de files d'attente de coordination à utiliser, puis cliquez sur **Connexion**.
3. Démarrez l'assistant **Créer un nouveau transfert de fichier géré** à l'aide de l'une des méthodes suivantes:
 - a) Cliquez avec le bouton droit de la souris sur le nom de l'un des noeuds suivants dans la vue Navigator : le gestionnaire de files d'attente de coordination approprié, **Modèles de transfert, Journal de transfert** ou **Transferts en attente**. Cliquez ensuite sur **Nouveau transfert** pour démarrer l'assistant.
 - b) Cliquez sur **Fichier > Nouveau > Autres > Assistants Managed File Transfer > Assistant Nouveau transfert**
4. Suivez les instructions des panneaux de l'assistant. Veillez à cocher la case **Activer le transfert de planification** et à entrer les détails de votre planification dans l'onglet **Planification** . Les transferts de fichiers planifiés démarrent dans la minute suivant l'heure de début de la planification, s'il n'y a aucun problème susceptible d'affecter le transfert. Par exemple, il peut y avoir des problèmes avec votre réseau ou votre agent qui empêchent le démarrage du transfert planifié. Une aide contextuelle est fournie pour chaque panneau. Pour accéder à l'aide contextuelle sur Windows, appuyez sur F1. Sous Linux, appuyez sur Ctrl+F1 ou sur Shift+F1.

Résultats

Pour plus d'informations sur les messages impliqués dans les transferts de fichiers planifiés, voir [Formats de message pour les transferts planifiés](#).

Utilisation des transferts de fichiers en attente

Vous pouvez afficher les transferts de fichiers planifiés en attente à partir du IBM MQ Explorer. La fenêtre **Transferts en attente** affiche tous les transferts en attente enregistrés auprès du gestionnaire de file d'attente de coordination auquel vous êtes actuellement connecté.

Pourquoi et quand exécuter cette tâche

Pour afficher le statut d'un transfert de fichiers planifié qui n'a pas encore démarré, procédez comme suit:

Procédure

1. Développez **Managed File Transfer** dans la vue Navigator . **Managed File Transfer Central** s'affiche dans la vue Contenu.
2. Tous vos gestionnaires de files d'attente de coordination sont affichés dans la vue Navigator . Développez le nom du gestionnaire de file d'attente de coordination que vous avez utilisé pour le transfert planifié. Si vous souhaitez modifier le gestionnaire de file d'attente de coordination auquel vous êtes connecté, cliquez avec le bouton droit de la souris sur le nom du gestionnaire de file d'attente de coordination à utiliser dans la vue Navigator , puis cliquez sur **Connexion**.
3. Cliquez sur **Transferts en attente**. La fenêtre **Transferts en attente** s'affiche dans la vue Contenu.
4. La fenêtre **Transferts en attente** affiche les détails suivants sur vos transferts de fichiers planifiés:
 - a) **Nom** Numéro du transfert de fichiers planifié. Ce numéro est automatiquement attribué.
 - b) **Source** Nom de l'agent source.
 - c) **Fichier source** Nom du fichier à transférer sur son système hôte.
 - d) **Destination** Nom de l'agent de destination.
 - e) **Fichier de destination** Nom du fichier après son transfert vers le système de destination.
 - f) **Début planifié (fuseau horaire sélectionné)** Date et heure auxquelles le transfert de fichier est planifié pour démarrer dans le fuseau horaire sélectionné de l'administrateur. Pour modifier le fuseau horaire affiché, cliquez sur **Fenêtre > Préférences > IBM MQ Explorer > Managed File Transfer** et sélectionnez un autre fuseau horaire dans la liste **Fuseau horaire** . Cliquez sur **OK**.
 - g) **Répéter toutes les** Si vous avez choisi de répéter le transfert planifié, l'intervalle spécifié que vous souhaitez répéter le transfert, exprimé sous la forme d'un nombre.
 - h) **Type de répétition** Si vous avez choisi de répéter le transfert planifié, le type d'intervalle de répétition que vous avez spécifié pour le transfert de fichiers. Le type peut être l'une des valeurs suivantes: minutes, heures, jours, semaines, mois ou années.
 - i) **Répéter jusqu'à** Si vous avez choisi de répéter le transfert planifié, les détails du moment où vous souhaitez que le transfert de fichiers répétitif s'arrête. Par exemple, une date et une heure spécifiées, ou après un nombre spécifié d'occurrences.

Résultats

Pour actualiser les éléments affichés dans la fenêtre **Transferts en attente** , cliquez sur le bouton Actualiser  dans la barre d'outils de la vue Contenu.

Pour annuler un transfert de fichiers en attente, cliquez avec le bouton droit de la souris sur le transfert concerné et cliquez sur **Annuler**. L'annulation d'un transfert supprime complètement la demande de transfert de fichier.

Déclenchement d'un transfert de fichiers

Vous pouvez définir certaines conditions de déclenchement sur un transfert de fichier qui doivent être vraies pour que ce transfert puisse avoir lieu. Si les conditions de déclenchement ne sont pas remplies, le transfert de fichier n'a pas lieu et un message de journal est éventuellement envoyé pour enregistrer le fait que le transfert n'a pas eu lieu. La demande de transfert de fichier est ensuite supprimée. Par

exemple, vous pouvez configurer un transfert de fichier qui n'a lieu que si un fichier nommé sur le système où se trouve l'agent source est supérieur à une taille spécifiée ou s'il existe un fichier nommé particulier sur le système où se trouve l'agent source. Vous pouvez configurer un transfert de fichier déclenché à partir de IBM MQ Explorer ou de la ligne de commande.

Pourquoi et quand exécuter cette tâche

Vous pouvez surveiller une ressource en permanence pour qu'une condition de déclenchement soit satisfaite. Pour plus d'informations sur la surveillance des ressources, voir: [«Surveillance des ressources MFT»](#), à la page 174.

Vous pouvez définir trois conditions de déclenchement différentes. Les conditions sont les suivantes :

- Si un fichier particulier existe sur le même système que l'agent source
- Si un fichier particulier n'existe pas sur le même système que l'agent source
- Si un fichier particulier dépasse une certaine taille sur le système où se trouve l'agent source (la taille peut être exprimée en octets, ko, Mo ou Go). Ces unités de mesure utilisent la convention 2¹⁰, par exemple 1 ko est égal à 1024 octets et 1 Mo est égal à 1024 ko.

Les types de déclenchement de la liste précédente peuvent être combinés de deux manières:

- Pour une condition unique, vous pouvez spécifier plusieurs fichiers sur le système où se trouve l'agent source. Cela déclenche le transfert si l'un des fichiers spécifiés remplit la condition (opérateur booléen OR).
- Vous pouvez spécifier plusieurs conditions. Cela déclenche le transfert uniquement si toutes les conditions sont remplies (opérateur booléen AND).

Vous pouvez également combiner un transfert déclenché avec un transfert planifié. Pour plus d'informations, voir [Création d'un transfert de fichiers planifié](#). Dans ce cas, les conditions de déclenchement sont évaluées au moment où la planification doit démarrer, ou pour une planification répétitive chaque fois que la planification doit démarrer.

Les transferts déclenchés ne sont pas pris en charge sur les agents de pont de protocole.

Pour créer un transfert de fichiers déclenché à l'aide de la ligne de commande, utilisez le paramètre **-tr** dans la commande [fteCreateTransfer](#).

Pour créer un transfert de fichier planifié à l'aide de l'assistant **Créer un transfert de fichier géré** dans IBM MQ Explorer, procédez comme suit:

Procédure

1. Dans la vue Navigator, cliquez sur **Transfert de fichier géré. Managed File Transfer Central** s'affiche dans la vue Contenu.
2. Tous vos gestionnaires de files d'attente de coordination sont affichés dans la vue Navigator. Développez le nom du gestionnaire de file d'attente de coordination que vous avez utilisé pour le transfert planifié. Si vous souhaitez modifier le gestionnaire de file d'attente de coordination auquel vous êtes connecté, cliquez avec le bouton droit de la souris sur le nom du gestionnaire de file d'attente de coordination à utiliser dans la vue Navigator, puis cliquez sur **Connexion**.
3. Démarrez l'assistant **Créer un nouveau transfert de fichier géré** à l'aide de l'une des méthodes suivantes:
 - a) Cliquez avec le bouton droit de la souris sur le nom de l'un des noeuds suivants dans la vue Navigator : le gestionnaire de files d'attente de coordination approprié, **Modèles de transfert, Journal de transfert** ou **Transferts en attente**. Cliquez ensuite sur **Nouveau transfert** pour ouvrir l'assistant.
 - b) Cliquez sur **Fichier > Nouveau > Autres > Assistants Managed File Transfer > Assistant Nouveau transfert**
4. Suivez les instructions des panneaux de l'assistant. Veillez à cocher la case **Activer le transfert déclenché** dans l'onglet **Déclencheurs** et à renseigner les zones de cet onglet pour configurer

le déclenchement. Une aide contextuelle est fournie pour chaque panneau. Pour accéder à l'aide contextuelle sur Windows, appuyez sur F1. Sous Linux, appuyez sur **Ctrl+F1** ou Shift+F1.

Surveillance des transferts de fichiers en cours

Vous pouvez surveiller un transfert de fichiers en cours à l'aide de l'onglet **Managed File Transfer-Current Transfer Progress** dans IBM MQ Explorer. Ce transfert de fichier peut être démarré à partir de IBM MQ Explorer ou de la ligne de commande. L'onglet affiche également la progression des transferts planifiés au point de début des transferts planifiés.

Pourquoi et quand exécuter cette tâche

Si vous souhaitez utiliser IBM MQ Explorer pour surveiller les transferts associés à un gestionnaire de files d'attente de coordination sur un système distant, suivez les instructions de la rubrique [«Configuration de IBM MQ Explorer pour la surveillance d'un gestionnaire de files d'attente de coordination distantes»](#), à la page 171 .

Les informations de transfert de fichiers précédentes ne sont pas conservées après l'arrêt et le redémarrage de IBM MQ Explorer. Au redémarrage, les informations sur les transferts passés sont effacées de l'onglet **Progression du transfert en cours** . Vous pouvez effacer les transferts terminés à l'aide de l'option **Supprimer les transferts terminés**  à tout moment lorsque IBM MQ Explorer est ouvert.

Procédure

Une fois que vous avez démarré un nouveau transfert de fichiers à l'aide de IBM MQ Explorer ou de la ligne de commande, vous pouvez surveiller la progression de votre transfert dans l'onglet **Progression du transfert en cours** . Les informations suivantes s'affichent pour chaque transfert en cours:

- a) **Source**. Nom de l'agent utilisé pour transférer le fichier à partir du système source.
- b) **Destination**. Nom de l'agent utilisé pour recevoir le fichier sur le système cible.
- c) **Fichier en cours**. Nom du fichier en cours de transfert. La partie du fichier individuel qui a déjà été transférée est affichée en B, KiB, MiB, GiB ou TiB avec la taille totale du fichier entre parenthèses. L'unité de mesure affichée dépend de la taille du fichier.
B est le nombre d'octets par seconde. KiB/s est le nombre de kibioctets par seconde, où 1 kibioctet est égal à 1024 octets. MiB/s correspond à des mébioctets par seconde, où 1 mébioctet est égal à 1 048 576 octets. GiB/s correspond à des gibioctets par seconde où 1 gibioctet est égal à 1 073 741 824 octets. TiB/s correspond à des tébioctets par seconde où 1 tébioctet est égal à 1 099 511 627 776 octets.
- d) **Numéro de fichier**. Si vous transférez plusieurs fichiers, ce nombre représente la distance par rapport au groupe total de fichiers transférés.
- e) **Progression**. La barre de progression indique le pourcentage d'achèvement du transfert de fichiers en cours.
- f) **Taux**. Débit de transfert du fichier dans KiB/s (kibioctets par seconde, où 1 kibioctet est égal à 1024 octets).
- g) **Démarré (fuseau horaire sélectionné)**. Heure à laquelle le transfert de fichier a commencé, présentée dans le fuseau horaire sélectionné de l'administrateur. Pour modifier le fuseau horaire affiché, cliquez sur **Fenêtre > Préférences > IBM MQ Explorer > Managed File Transfer** et sélectionnez un autre fuseau horaire dans la liste **Fuseau horaire** : . Cliquez sur **OK**.

Si le transfert passe à l'état de récupération lors du transfert du fichier, l'heure de début est mise à jour pour refléter l'heure à laquelle le transfert de fichier a repris.

Résultats

Cet onglet actualise régulièrement ses informations automatiquement, mais pour forcer une vue actualisée de ce qui est affiché dans l'onglet **Progression du transfert en cours**, cliquez sur **Actualiser**  dans la barre d'outils de la vue de contenu.

Pour supprimer des transferts de fichiers à partir de l'onglet **Progression du transfert en cours**, cliquez sur **Supprimer les transferts terminés**  dans la barre d'outils de la vue Contenu. Cliquez sur ce bouton pour supprimer les détails du transfert de fichiers de l'onglet uniquement ; il n'arrête pas ou n'annule pas un transfert en cours ou planifié.

Si vous souhaitez revenir à l'onglet **Progression du transfert en cours** après sa fermeture, vous pouvez l'afficher en cliquant sur **Fenêtre > Afficher la vue > Autre > Autre > Transfert de fichiers géré-Progression du transfert en cours**. Cliquez sur **OK**.

Que faire ensuite

De plus, il est possible de développer des applications pour la surveillance du transfert de fichiers personnalisé. Pour ce faire, vous pouvez créer un abonnement à la rubrique d'administration Managed File Transfer appropriée (à l'aide d'un programme ou d'un administrateur), et l'application de surveillance peut alors recevoir des publications d'activité de transfert de fichier Managed File Transfer sur la rubrique. Pour plus d'informations sur la rubrique relative à l'abonnement et le format des messages de publication, voir [«Exemples de message de progression de transfert de fichier»](#), à la page 770.

Tâches associées

[«Configuration de IBM MQ Explorer pour la surveillance d'un gestionnaire de files d'attente de coordination distantes»](#), à la page 171

Utilisez IBM MQ Explorer pour surveiller les transferts de fichiers associés à un gestionnaire de files d'attente de coordination s'exécutant sur un système distant. Dans IBM WebSphere MQ 7.5 ou version ultérieure, vous avez besoin d'un système capable d'exécuter IBM MQ Explorer. Le composant IBM MQ Explorer doit être installé pour pouvoir se connecter au gestionnaire de files d'attente de coordination à distance.

[«Affichage du statut des transferts de fichiers dans le journal de transfert»](#), à la page 172

Vous pouvez afficher les détails des transferts de fichiers à l'aide du **Journal de transfert** dans IBM MQ Explorer. Il peut s'agir de transferts démarrés à partir de la ligne de commande ou du IBM MQ Explorer. Vous pouvez également personnaliser les éléments affichés dans le **Journal de transfert**.

Configuration de IBM MQ Explorer pour la surveillance d'un gestionnaire de files d'attente de coordination distantes

Utilisez IBM MQ Explorer pour surveiller les transferts de fichiers associés à un gestionnaire de files d'attente de coordination s'exécutant sur un système distant. Dans IBM WebSphere MQ 7.5 ou version ultérieure, vous avez besoin d'un système capable d'exécuter IBM MQ Explorer. Le composant IBM MQ Explorer doit être installé pour pouvoir se connecter au gestionnaire de files d'attente de coordination à distance.

Pourquoi et quand exécuter cette tâche

Hypothèses: droits permettant de se connecter au gestionnaire de files d'attente de coordination distantes en configurant le gestionnaire de files d'attente de manière à autoriser les connexions distantes.

Pour plus d'informations sur la configuration, voir [«Connexion à un gestionnaire de files d'attente en mode client avec authentification de canal»](#), à la page 54 et [«Droits de gestion pour les ressources spécifiques à MFT»](#), à la page 361.

Pour surveiller les gestionnaires de files d'attente et les transferts de fichiers entre les agents sur un système qui n'exécute pas Windows ou Linux, configurez IBM MQ Explorer pour qu'il se connecte au système distant en procédant comme suit:

Procédure

1. Démarrez le IBM MQ Explorer local.
2. Lorsque IBM MQ Explorer est chargé, cliquez avec le bouton droit de la souris sur le dossier **Managed File Transfer** et sélectionnez **Nouvelle configuration**.
3. Exécutez l'assistant en sélectionnant le gestionnaire de files d'attente de coordination et de commandes, puis définissez un nom pour la configuration.
4. Cliquez sur **Terminer** pour terminer la définition.
5. Une fois la définition terminée, cliquez avec le bouton droit de la souris sur la définition et sélectionnez **Connecter**.

Résultats

A présent, démarrez IBM MQ Explorer et utilisez-le pour surveiller l'activité de transfert pour le réseau Managed File Transfer associé au gestionnaire de files d'attente de coordination.

Tâches associées

«Surveillance des transferts de fichiers en cours», à la page 170

Vous pouvez surveiller un transfert de fichiers en cours à l'aide de l'onglet **Managed File Transfer-Current Transfer Progress** dans IBM MQ Explorer. Ce transfert de fichier peut être démarré à partir de IBM MQ Explorer ou de la ligne de commande. L'onglet affiche également la progression des transferts planifiés au point de début des transferts planifiés.

«Affichage du statut des transferts de fichiers dans le journal de transfert», à la page 172

Vous pouvez afficher les détails des transferts de fichiers à l'aide du **Journal de transfert** dans IBM MQ Explorer. Il peut s'agir de transferts démarrés à partir de la ligne de commande ou du IBM MQ Explorer. Vous pouvez également personnaliser les éléments affichés dans le **Journal de transfert**.

Affichage du statut des transferts de fichiers dans le journal de transfert

Vous pouvez afficher les détails des transferts de fichiers à l'aide du **Journal de transfert** dans IBM MQ Explorer. Il peut s'agir de transferts démarrés à partir de la ligne de commande ou du IBM MQ Explorer. Vous pouvez également personnaliser les éléments affichés dans le **Journal de transfert**.

Procédure

1. Développez **Managed File Transfer** dans la vue Navigator , puis développez le nom du gestionnaire de files d'attente de coordination dont vous souhaitez afficher le journal de transfert.
2. Cliquez sur **Journal de transfert** dans la vue Navigator . Le **journal de transfert** s'affiche dans la vue Contenu.
3. La fenêtre **Journal de transfert** affiche les détails suivants sur vos transferts de fichiers:
 - a) **Source** Nom de l'agent sur le système où se trouve le fichier source.
 - b) **Destination** Nom de l'agent sur le système vers lequel vous souhaitez transférer le fichier.
 - c) **Etat d'achèvement** Statut du transfert de fichier. L'état peut être l'une des valeurs suivantes: "Démarré", "En cours", "Réussite", "Partiellement réussi", "Annulé" ou "Echec".
 - d) **Propriétaire** ID utilisateur sur l'hôte qui a soumis la demande de transfert.
 - e) **Démarré (fuseau horaire sélectionné)** Date et heure auxquelles la demande de transfert de fichier a été acceptée par l'agent Managed File Transfer , présentées dans le fuseau horaire sélectionné de l'administrateur. Pour modifier le fuseau horaire affiché, cliquez sur **Fenêtre > Préférences > IBM MQ Explorer > Managed File Transfer** et sélectionnez un autre fuseau horaire dans la liste **Fuseau horaire** . Cliquez sur **OK**.
 - f) **Etat enregistré (fuseau horaire sélectionné)** (Cette colonne n'est pas affichée par défaut. Vous pouvez choisir d'afficher la colonne à l'aide de la fenêtre **Configurer les colonnes du journal de transfert**  . Heure et date d'enregistrement de l'état d'achèvement, dans le fuseau horaire sélectionné par l'administrateur.

- g) **Nom du travail** Identificateur spécifié par l'utilisateur à l'aide du paramètre **-jn** de **fteCreateTransfer** ou dans un script Ant
- h) **ID transfert** Identificateur unique du transfert de fichier.
- i) **Connect: Direct** Les détails relatifs au **numéro de processus**, au **nom de processus**, au **noeud principal**, au **noeud secondaire**, au **type de source** et au **type de destination** sont répertoriés.

Résultats

Remarque : Le format interne du journal de transfert a été modifié dans IBM MQ 8.0.0 Fix Pack 1 pour l'APAR IC99545. Par conséquent, si un IBM MQ Explorer est mis à niveau vers 8.0.0.1 ou une version ultérieure, puis restauré vers 8.0.0.0, aucun code XML d'audit n'est affiché pour les transferts qui ont eu lieu alors que IBM MQ Explorer était à 8.0.0.1. Le panneau XML de la fenêtre **Propriétés** de ces transferts contient une zone de texte vide.

Pour afficher plus de détails sur un transfert terminé, développez le transfert qui vous intéresse en cliquant sur le signe plus (+). Vous pouvez ensuite voir tous les noms de fichier source et de destination inclus dans ce transfert. Toutefois, si le transfert est en cours et se compose de nombreux fichiers, vous pouvez afficher uniquement les fichiers qui ont déjà été transférés jusqu'à présent.

Pour actualiser les éléments affichés dans le **Journal de transfert**, cliquez sur le bouton **Actualiser**  dans la barre d'outils de la vue de contenu. Les informations de transfert de fichier dans le journal de transfert restent dans le journal après l'arrêt et le redémarrage de IBM MQ Explorer. Si vous souhaitez supprimer du journal tous les transferts de fichiers terminés, cliquez sur **Supprimer les transferts terminés**  dans la barre d'outils de la vue Contenu.

Pour supprimer un transfert de fichier individuel terminé du journal, cliquez avec le bouton droit de la souris sur le transfert et cliquez sur **Supprimer**. Si vous supprimez un transfert, il n'arrête pas ou n'annule pas un transfert qui est en cours ou qui a été planifié ; vous supprimez uniquement les données d'historique stockées.

Pour copier l'identificateur unique d'un transfert dans le presse-papiers, cliquez avec le bouton droit de la souris sur ce transfert et cliquez sur **Copier l'ID**.

Les métadonnées et le XML d'audit complet du transfert sont disponibles dans le menu contextuel, sous l'action **Propriétés**.

Concepts associés

«Option de délai d'attente pour les transferts de fichiers en cours de récupération», à la page 253
Vous pouvez définir la durée, en secondes, pendant laquelle un agent source tente de récupérer un transfert de fichiers bloqué. Si le transfert n'aboutit pas lorsque l'agent atteint le délai d'attente pour l'intervalle entre les nouvelles tentatives, le transfert échoue.

Tâches associées

«Surveillance des transferts de fichiers en cours», à la page 170

Vous pouvez surveiller un transfert de fichiers en cours à l'aide de l'onglet **Managed File Transfer-Current Transfer Progress** dans IBM MQ Explorer. Ce transfert de fichier peut être démarré à partir de IBM MQ Explorer ou de la ligne de commande. L'onglet affiche également la progression des transferts planifiés au point de début des transferts planifiés.

«Configuration du journal de transfert», à la page 173

Vous pouvez configurer les informations qui sont affichées et la façon dont les informations sont affichées dans le **journal de transfert** dans le IBM MQ Explorer.

Configuration du journal de transfert

Vous pouvez configurer les informations qui sont affichées et la façon dont les informations sont affichées dans le **journal de transfert** dans le IBM MQ Explorer.

Pourquoi et quand exécuter cette tâche

Pour réorganiser l'ordre des colonnes dans le **Journal de transfert**, cliquez sur le titre de la colonne à déplacer et faites glisser la colonne vers sa nouvelle position. Le nouvel ordre des colonnes est conservé uniquement jusqu'à ce que vous arrêtez et redémarriez IBM MQ Explorer.

Pour filtrer les entrées du **journal de transfert**, entrez une chaîne dans la zone **Filtrer les entrées de journal affichées**. Pour restaurer toutes les entrées du journal, supprimez la chaîne que vous avez entrée dans la zone. Vous pouvez utiliser n'importe quelle expression régulière Java valide dans cette zone. Pour plus d'informations, voir la section «Expressions régulières utilisées par MFT», à la page 679.

Pour personnaliser les colonnes affichées dans le journal de transfert, utilisez **Configurer les colonnes du journal de transfert** . Procédez comme suit pour démarrer et utiliser la fenêtre **Configurer les colonnes du journal de transfert**.

Procédure

1. Vérifiez que le **journal de transfert** est ouvert dans la vue Contenu. Cliquez sur **Configurer les colonnes du journal de transfert**  dans la barre d'outils de la vue de contenu. La fenêtre **Configurer les colonnes du journal de transfert** s'ouvre.
2. Pour personnaliser votre vue du **journal de transfert**, cochez ou décochez des cases individuelles pour les colonnes que vous souhaitez afficher ou masquer. Vous pouvez cliquer sur **Sélectionner tout**, puis sur **OK** pour sélectionner toutes les cases à cocher ou sur **Désélectionner tout**, puis sur **OK** pour désélectionner toutes les cases.

Tâches associées

«Surveillance des transferts de fichiers en cours», à la page 170

Vous pouvez surveiller un transfert de fichiers en cours à l'aide de l'onglet **Managed File Transfer-Current Transfer Progress** dans IBM MQ Explorer. Ce transfert de fichier peut être démarré à partir de IBM MQ Explorer ou de la ligne de commande. L'onglet affiche également la progression des transferts planifiés au point de début des transferts planifiés.

«Affichage du statut des transferts de fichiers dans le journal de transfert», à la page 172

Vous pouvez afficher les détails des transferts de fichiers à l'aide du **Journal de transfert** dans IBM MQ Explorer. Il peut s'agir de transferts démarrés à partir de la ligne de commande ou du IBM MQ Explorer. Vous pouvez également personnaliser les éléments affichés dans le **Journal de transfert**.

Surveillance des ressources MFT

Vous pouvez surveiller les ressources Managed File Transfer ; par exemple, une file d'attente ou un répertoire. Lorsqu'une condition est satisfaite sur cette ressource, le moniteur de ressources démarre une tâche, telle qu'un transfert de fichier. Vous pouvez créer un moniteur de ressources à l'aide de la commande **fteCreateMonitor** ou de la vue **Moniteurs** dans le plug-in Managed File Transfer pour IBM MQ Explorer.

Pourquoi et quand exécuter cette tâche

La surveillance des ressources Managed File Transfer utilise la terminologie suivante:

moniteur

Processus qui interroge une ressource (telle qu'un répertoire ou une file d'attente) à un intervalle régulier prédéfini pour voir si le contenu de la ressource a changé. Si tel est le cas, le contenu est comparé à l'ensemble des conditions de ce moniteur. S'il existe une correspondance, la tâche de ce moniteur est démarrée.

ressource

La ressource système que le moniteur examine chaque intervalle d'interrogation à comparer aux conditions de déclenchement. Les files d'attente, les répertoires ou les structures de répertoire imbriquées peuvent être la ressource contrôlée.

condition

Expression évaluée (généralement par rapport au contenu de la ressource surveillée). Si l'expression a pour résultat la valeur true, la condition contribue à la condition de déclenchement globale.

Condition de déclenchement

Condition globale, qui est remplie lorsque toutes les conditions sont remplies. Lorsque la condition de déclenchement est satisfaite, la tâche peut se poursuivre.

tâche

Opération démarrée lorsque la condition de déclenchement ou l'ensemble de conditions est satisfait. Les tâches prises en charge sont le transfert de fichiers et l'appel de commande.

fichier de déclenchement

Fichier placé dans un répertoire contrôlé pour indiquer qu'une tâche (généralement un transfert) peut commencer. Par exemple, il peut indiquer que tous les fichiers à traiter sont arrivés dans un emplacement connu et peuvent être transférés ou traités d'une autre manière. Le nom du fichier de déclenchement peut être utilisé pour spécifier les fichiers à transférer à l'aide de la substitution de variable. Pour plus d'informations, voir la section «Personnalisation des tâches du moniteur de ressources MFT avec la substitution de variables», à la page 185.

Le fichier de déclenchement est également appelé fichier prêt ou fichier go. Toutefois, dans cette documentation, il est toujours appelé fichier de déclenchement.

La surveillance des ressources n'est pas prise en charge sur les agents de pont de protocole ou sur les Connect:Direct agents de pont.

Concepts de surveillance des ressources MFT

Présentation des concepts clés de la fonction de surveillance des ressources Managed File Transfer .

Moniteurs

Le moniteur de ressources est associé à un agent Managed File Transfer et n'est actif que lorsque cet agent est démarré. Lorsque l'agent de surveillance s'arrête, il en est de même pour le moniteur. Si l'agent est déjà démarré lors de la création du moniteur, ce dernier démarre immédiatement. L'agent de surveillance doit également être l'agent source de la tâche lancée par le moniteur.

Les noms de moniteur doivent être uniques au sein de leur agent. Le nom du moniteur doit comporter au moins un caractère et ne doit pas contenir d'astérisque (*), de pourcentage (%) ou de point d'interrogation (?). La casse des noms de moniteur fournis est ignorée et le nom de moniteur est converti en majuscules. Si vous tentez de créer un moniteur avec un nom déjà présent, la demande est ignorée et la tentative est consignée dans la rubrique du journal du moniteur.

Il n'y a aucune restriction sur le nombre de moniteurs pouvant être créés sur un agent et tous exécutés avec la même priorité. Tenez compte des implications du chevauchement des ressources surveillées, des conditions de déclenchement conflictuelles et de la fréquence d'interrogation des ressources.

Le chevauchement des moniteurs de ressources peut entraîner:

- Conflit possible sur le / les emplacement (s) source (s).
- Demandes de transfert en double possibles pour les mêmes éléments source.
- Erreurs ou échecs inattendus pour les transferts dus à des conflits d'éléments source.

Si plusieurs moniteurs analysent le même emplacement et peuvent se déclencher sur les mêmes éléments, vous pouvez potentiellement vous retrouver avec le problème de deux moniteurs différents soumettant des demandes de transfert gérées pour le même élément.

Les moniteurs de ressources regardent le contenu des ressources après chaque période d'intervalle d'interrogation. Le contenu de la ressource est comparé aux conditions de déclenchement et si ces conditions sont satisfaites, la tâche associée au moniteur est appelée.

La tâche est démarrée de manière asynchrone. S'il existe une condition de correspondance et que la tâche est démarrée, le moniteur continue d'interroger les modifications apportées au contenu de la ressource. Par exemple, si une correspondance est trouvée parce qu'un fichier appelé `reports.go` est

arrivé dans un répertoire contrôlé, la tâche est démarrée une seule fois. Lors de l'intervalle d'interrogation suivant, même si le fichier existe encore, la tâche n'est pas redémarrée. Toutefois, si le fichier est supprimé, puis placé à nouveau dans le répertoire, ou si le fichier est mis à jour (de sorte que l'attribut de date de dernière modification est modifié), la vérification de la condition de déclenchement suivante entraîne un nouvel appel de la tâche.

Ressources

Les moniteurs de ressources dans Managed File Transfer peuvent interroger le contenu des deux types de ressource suivants:

Répertoires ou structures de répertoire imbriquées

Un scénario courant consiste à surveiller un répertoire pour détecter la présence d'un fichier de déclenchement. Une application externe peut traiter plusieurs fichiers et les placer dans un répertoire source connu. Lorsque l'application a terminé son traitement, elle indique que les fichiers sont prêts à être transférés, ou à être traités d'une autre manière, en plaçant un fichier de déclenchement dans un emplacement surveillé. Le fichier de déclenchement peut être détecté par un moniteur de ressources Managed File Transfer et le transfert de ces fichiers du répertoire source vers un autre Managed File Transfer Agent est lancé.

Par défaut, le répertoire spécifié est surveillé. Pour examiner également les sous-répertoires, définissez le niveau de récursivité dans la commande **fteCreateTransfer**.

Voici deux exemples de surveillance d'un répertoire:

- Surveillez un fichier de déclenchement (par exemple, `trigger.file`), puis transférez un caractère générique (par exemple, `*.zip`).
- Surveillez `*.zip`, puis transférez `${FilePath}` (par exemple, le fichier qui a déclenché le transfert). Pour plus d'informations sur la substitution de variables, voir [«Personnalisation des tâches du moniteur de ressources MFT avec la substitution de variables»](#), à la page 185.

Remarque : Ne créez pas de moniteur qui surveille `*.zip`, puis transfère `*.zip`. Le moniteur tente de démarrer un transfert de `*.zip` pour chaque fichier `.zip` de votre système. C'est-à-dire que le moniteur génère * nombre de transferts pour `*.zip`.

Pour obtenir un exemple de création d'un moniteur de ressources afin de surveiller un répertoire, voir [«Surveillance d'un répertoire et utilisation de la substitution de variable»](#), à la page 183.

Files d'attente IBM MQ

Un exemple de surveillance d'une file d'attente est qu'une application externe peut générer des messages et les placer dans une file d'attente connue avec le même ID de groupe. Lorsque l'application a terminé d'insérer des messages dans la file d'attente, elle indique que le groupe est terminé. Le groupe complet de messages peut être détecté par un moniteur de ressources Managed File Transfer et le transfert du groupe de messages de la file d'attente source vers un fichier est lancé. Pour un exemple de création d'un moniteur de ressources pour surveiller une file d'attente, voir [«Exemple: configuration d'une ressource MFT»](#), à la page 184.

Remarque : Vous ne pouvez spécifier qu'un seul moniteur par file d'attente. Si vous spécifiez plusieurs moniteurs pour interroger une file d'attente IBM MQ, un comportement imprévisible se produit.

Les fichiers de surveillance ne sont pas pris en charge.

Conditions de déclenchement

La condition est remplie lorsque la ressource contient une valeur qui correspond à une autre chaîne ou à un autre modèle. Les conditions peuvent être l'une des suivantes:

- Correspondance sur le nom de fichier (modèle)
- Aucune correspondance sur le nom de fichier (modèle)
- Taille de fichier
- Correspond si la taille du fichier reste la même pour un certain nombre d'interrogations

La correspondance de nom de fichier peut être exprimée comme suit:

- Correspondance de chaîne exacte
- Correspondance avec des caractères génériques simples, comme décrit dans [«Utilisation de caractères génériques avec MFT»](#), à la page 673
- Correspondance d'expression régulière

Les noms de fichier peuvent également être exclus de la mise en correspondance des noms de fichier à l'aide d'un caractère générique ou d'une expression régulière Java qui identifie les noms de fichier qui ne sont jamais mis en correspondance.

Lorsqu'un fichier correspondant est détecté, son horodatage de dernière modification est conservé. Si les interrogations suivantes détectent que le fichier a été modifié, la condition de déclenchement est remplie à nouveau et la tâche est démarrée. Si la condition est de détecter lorsqu'un fichier n'existe pas, si aucun fichier du répertoire contrôlé ne correspond au modèle de nom de fichier, la tâche est démarrée. Si un fichier est ensuite ajouté au répertoire qui correspond au modèle de nom de fichier, la tâche est démarrée uniquement si le fichier est ensuite supprimé.

Tâches

Managed File Transfer prend en charge les deux types de tâche suivants que vous pouvez configurer pour qu'ils soient démarrés par les moniteurs de ressources:

- Transfert de fichier
- Commande

Les tâches de transfert de fichiers sont définies de la même manière que tout autre transfert de fichiers. Une méthode utile pour générer le code XML de tâche requis par un moniteur consiste à exécuter la commande `fteCreateTransfer` avec le paramètre `-gt`. Cette commande génère une définition de tâche en tant que document XML, y compris la spécification de transfert. Vous transmettez ensuite le nom du document XML de tâche comme valeur du paramètre `-mt` dans la commande `fteCreateMonitor`. Lorsque le `fteCreateMonitor` est exécuté, il lit le document XML de la tâche. Une fois le `fteCreateMonitor` exécuté, les modifications apportées au fichier XML de tâche ne sont pas utilisées par le moniteur.

Les tâches de commande peuvent exécuter des scripts Ant, appeler des programmes exécutables ou exécuter des travaux JCL. Pour plus d'informations, voir [Configuration des tâches de surveillance pour appeler des commandes et des scripts](#).

Lorsque vous utilisez une tâche de transfert de fichiers, vous pouvez sélectionner le nombre de conditions de déclenchement qui sont regroupées par lots dans une tâche. Par défaut, une condition de déclenchement démarre une tâche. Vous pouvez exécuter la commande `fteCreateMonitor` avec l'option `-bs` pour sélectionner le nombre de conditions de déclenchement regroupées par lots dans une tâche.

Sauvegarde et restauration des moniteurs de ressources

Vous pouvez sauvegarder les moniteurs de ressources que vous avez déjà définis afin de pouvoir les réutiliser ultérieurement. Il existe différentes options que vous pouvez utiliser comme suit:

- Utilisez la commande `fteCreateMonitor` avec le paramètre `-ox` pour exporter une configuration de moniteur de ressources dans un fichier XML et avec le paramètre `-ix` pour restaurer un moniteur de ressources en important la configuration de moniteur de ressources à partir d'un fichier XML.
- Utilisez la commande `fteListMonitors` avec `-ox` pour exporter la définition d'un moniteur de ressources unique dans un fichier XML.
- **V 9.0.5** Depuis IBM MQ 9.0.5, utilisez la commande `fteListMonitors` avec `-od` pour exporter plusieurs définitions de moniteur de ressources vers un répertoire spécifié. Chaque définition de moniteur de ressources est sauvegardée dans un fichier XML distinct. Vous pouvez également utiliser l'option `-od` pour exporter une définition de moniteur de ressources unique vers un répertoire spécifié.

Pour plus d'informations, voir la section [«Sauvegarde et restauration des moniteurs de ressources MFT»](#), à la page 199.

Consignation du moniteur de ressources

V 9.0.3

Depuis la IBM MQ 9.0.3, Managed File Transfer inclut la journalisation du moniteur de ressources. Pour plus d'informations, voir la section [«Consignation des moniteurs de ressources MFT»](#), à la page 196.

Concepts associés

[«Personnalisation des tâches du moniteur de ressources MFT avec la substitution de variables»](#), à la page 185

Lorsque les conditions de déclenchement d'un moniteur de ressources actif sont satisfaites, la tâche définie est appelée. En plus d'appeler la tâche de transfert ou de commande avec le même agent de destination ou le même nom de fichier de destination à chaque fois, vous pouvez également modifier la définition de la tâche lors de l'exécution. Pour ce faire, vous devez insérer des noms de variable dans le fichier XML de définition de tâche. Lorsque le moniteur détermine que les conditions de déclenchement sont satisfaites et que la définition de tâche contient des noms de variable, il remplace les noms de variable par les valeurs de variable, puis appelle la tâche.

Tâches associées

[«Configuration des tâches de surveillance MFT pour lancer des commandes et des scripts»](#), à la page 178

Les moniteurs de ressources ne sont pas limités à l'exécution de transferts de fichiers en tant que tâche associée. Vous pouvez également configurer le moniteur pour qu'il appelle d'autres commandes à partir de l'agent de surveillance, notamment des programmes exécutables, des scripts Ant ou des travaux JCL. Pour appeler des commandes, éditez le fichier XML de définition de tâche de contrôle pour inclure un ou plusieurs éléments de commande avec les paramètres d'appel de commande correspondants, tels que les arguments et les propriétés.

[«Exemple: configuration d'une ressource MFT»](#), à la page 184

Vous pouvez spécifier une file d'attente IBM MQ comme ressource à surveiller par un moniteur de ressources à l'aide du paramètre **-mq** avec la commande **fteCreateMonitor**.

[«Surveillance d'une file d'attente et utilisation de la substitution de variable»](#), à la page 191

Vous pouvez surveiller une file d'attente et transférer des messages de la file d'attente surveillée vers un fichier à l'aide de la commande **fteCreateMonitor**. La valeur de toute propriété de message IBM MQ dans le premier message à lire dans la file d'attente surveillée peut être remplacée dans la définition XML de tâche et utilisée pour définir le comportement de transfert.

Référence associée

[«fteCreateMonitor: créer un moniteur de ressources MFT»](#), à la page 424

La commande **fteCreateMonitor** crée et démarre un nouveau moniteur de ressources à partir de la ligne de commande. Vous pouvez surveiller une ressource (par exemple, le contenu d'un répertoire) à l'aide de Managed File Transfer de sorte que lorsqu'une condition de déclenchement est remplie, une tâche spécifiée, telle qu'un transfert de fichier, est démarrée.

[«fteListMoniteurs: répertorie les moniteurs de ressources MFT»](#), à la page 487

La commande **fteListMonitors** permet de répertorier tous les moniteurs de ressources existants dans un réseau Managed File Transfer à l'aide de la ligne de commande.

[«Moniteur fteDelete: suppression d'un moniteur de ressources MFT»](#), à la page 477

La commande **fteDeleteMonitor** permet d'arrêter et de supprimer un moniteur de ressources Managed File Transfer existant à l'aide de la ligne de commande. Exécutez cette commande sur l'agent de surveillance des ressources.

Configuration des tâches de surveillance MFT pour lancer des commandes et des scripts

Les moniteurs de ressources ne sont pas limités à l'exécution de transferts de fichiers en tant que tâche associée. Vous pouvez également configurer le moniteur pour qu'il appelle d'autres commandes à partir de l'agent de surveillance, notamment des programmes exécutables, des scripts Ant ou des travaux JCL. Pour appeler des commandes, éditez le fichier XML de définition de tâche de contrôle pour inclure un ou plusieurs éléments de commande avec les paramètres d'appel de commande correspondants, tels que les arguments et les propriétés.

Pourquoi et quand exécuter cette tâche

Le chemin d'accès au programme exécutable, au script Ant ou au travail JCL que vous souhaitez que l'agent de surveillance appelle doit être inclus dans le fichier `commandPath` de l'agent de surveillance. Pour plus d'informations sur la propriété de chemin de commande, voir «[Propriété `commandPath` MFT](#)», à la page 374.

Vous pouvez créer le document XML de définition de tâche de l'une des manières suivantes:

- Créez le document XML de définition de tâche manuellement en fonction du schéma `FileTransfer.xsd`. Pour plus d'informations, voir la section «[Créer manuellement le XML de définition de tâche en fonction du schéma](#)», à la page 179.
- Editez le document XML généré par le paramètre `fteCreateTransfer -gt` comme base de votre définition de tâche. Pour plus d'informations, voir la section «[Création d'un document de définition de tâche en modifiant un document généré](#)», à la page 181.

Que vous souhaitiez une tâche de transfert ou une tâche de commande, la définition de tâche doit commencer par un élément racine `<request>`. L'élément enfant de `<request>` doit être `<managedTransfer>` ou `<managedCall>`. Vous choisissez généralement `<managedCall>` lorsqu'il n'y a qu'une seule commande ou un seul script à exécuter, et `<managedTransfer>` si vous souhaitez que la tâche inclue un transfert de fichier et éventuellement jusqu'à quatre appels de commande.

Créer manuellement le XML de définition de tâche en fonction du schéma

Pourquoi et quand exécuter cette tâche

Vous pouvez créer manuellement un fichier XML de définition de tâche en fonction du schéma `FileTransfer.xsd`. Ce schéma se trouve dans le fichier `MQ_INSTALLATION_PATH/mqft/samples/schema`. Pour plus d'informations sur ce schéma, voir «[Format de message de demande de transfert de fichier](#)», à la page 804.

Exemple

L'exemple suivant illustre un exemple de document XML de définition de tâche sauvegardé sous `cleanuptask.xml`, qui utilise l'élément `<managedCall>` pour appeler un script Ant appelé `RunCleanup.xml`. Le script Ant `RunCleanup.xml` doit se trouver dans le chemin de commande `commandPath` de l'agent de surveillance.

```
<?xml version="1.0" encoding="UTF-8"?>
<request version="4.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedCall>
    <originator>
      <hostName>hostName</hostName>
      <userID>userID</userID>
      <mqmdUserID>mqmdUserID</mqmdUserID>
    </originator>
    <agent QMGR="QM1" agent="AGENT1"/>
    <reply QMGR="QM1">reply</reply>
    <transferSet priority="1">
      <metaDataSet>
        <metaData key="name1">value1</metaData>
      </metaDataSet>
      <call>
        <command name="RunCleanup.xml" type="antscript" retryCount="2"
          retryWait="30" successRC="0">
          <target>check_exists</target>
          <target>copy_to_archive</target>
          <target>rename_temps</target>
          <target>delete_files</target>
          <property name="trigger.filename" value="{FileName}"/>
          <property name="trigger.path" value="{FilePath}"/>
        </command>
      </call>
    </transferSet>
  </managedCall>
</request>
```

```
</managedCall>  
</request>
```

L'élément `<agent>` spécifie le Managed File Transfer Agent qui est configuré avec le script Ant nommé sur son chemin `commandPath`.

La structure `<call><command>` . . . définit l'exécutable ou le script que vous souhaitez exécuter. La commande prend un attribut `type` facultatif qui peut avoir l'une des valeurs suivantes:

antscript

Exécutez un script Ant dans une machine virtuelle Java distincte.

exécutable

Appelez un programme exécutable.

JCL

Appelez un travail JCL.

Si vous omettez l'attribut `type` , la valeur par défaut `executable` est utilisée.

L'attribut `name` indique le nom du script Ant, de l'exécutable ou du travail JCL que vous souhaitez exécuter, sans aucune information de chemin. L'agent recherche le script ou le programme dans les emplacements spécifiés par la propriété `commandPath` dans le fichier `agent.properties` de l'agent.

L'attribut `retrycount` indique le nombre de tentatives d'appel du programme si le programme ne renvoie pas de code retour de réussite. La valeur affectée à cet attribut ne doit pas être négative. Si vous ne spécifiez pas l'attribut `retrycount` , la valeur par défaut zéro est utilisée.

L'attribut `retrywait` indique le délai d'attente, en secondes, avant une nouvelle tentative d'appel du programme. La valeur affectée à cet attribut ne doit pas être négative. Si vous ne spécifiez pas l'attribut `retrywait` , la valeur par défaut zéro est utilisée.

L'attribut `successrc` est une expression utilisée pour déterminer quand l'appel de programme s'exécute correctement. Le code retour du processus pour la commande est évalué à l'aide de cette expression. La valeur peut être composée d'une ou de plusieurs expressions combinées avec un caractère de barre verticale (|) pour signifier la valeur booléenne OR ou une perluète (&) pour indiquer la valeur booléenne AND. Chaque expression peut être l'un des types d'expression suivants:

- Nombre indiquant un test d'égalité entre le code retour du processus et le nombre.
- Nombre précédé d'un caractère supérieur à (>) pour indiquer un test supérieur à entre le nombre et le code retour du processus.
- Nombre précédé d'un caractère inférieur à (<) pour indiquer un test inférieur à entre le nombre et le code retour du processus.
- Nombre précédé d'un point d'exclamation (!) pour indiquer un test non égal à entre le nombre et le code retour du processus. Par exemple: `> 2 & < 7 & ! 5 | 0 | 14` est interprété comme les codes retour suivants ayant abouti: 0, 3, 4, 6, 14. Tous les autres codes retour sont interprétés comme n'ayant pas abouti.

Si vous ne spécifiez pas l'attribut `successrc` , la valeur par défaut zéro est utilisée. Cela signifie que la commande est considérée comme ayant abouti si, et seulement si, elle renvoie un code de zéro.

Pour un script Ant, vous spécifiez généralement les éléments `<target>` et `<property>` . Les valeurs de l'élément `<target>` doivent correspondre aux noms de cible dans le script Ant.

Pour les programmes exécutables, vous pouvez spécifier des éléments `<argument>` . Les éléments d'argument imbriqués spécifient les arguments à transmettre au programme appelé dans le cadre de l'appel de programme. Les arguments de programme sont générés à partir des valeurs spécifiées par les éléments d'argument dans l'ordre dans lequel les éléments d'argument sont détectés. Vous pouvez spécifier zéro ou plusieurs éléments d'argument en tant qu'éléments imbriqués d'un appel de programme.

L'administrateur définit et démarre le moniteur comme normal à l'aide du document XML de définition de tâche qui inclut l'élément `<managedCall>`. Exemple :

```
fteCreateMonitor -ma AGENT1 -mm QM1 -md /monitored -mn MONITOR01 -mt
/tasks/cleanuptask.xml -pi 30 -pu seconds -tr match,*go
```

Le chemin d'accès au document XML de définition de transfert doit se trouver sur le système de fichiers local à partir duquel vous exécutez la commande **fteCreateMonitor** (dans cet exemple, `/tasks/cleanuptask.xml`). Le document `cleanuptask.xml` est utilisé pour créer le moniteur de ressources uniquement. Toutes les tâches référencées par le document `cleanuptask.xml` (scripts Ant ou travaux JCL) doivent se trouver dans le chemin de commande de l'agent de surveillance. Lorsque la condition de déclenchement du moniteur est satisfaite, toutes les variables du fichier XML de définition de tâche sont remplacées par les valeurs réelles du moniteur. Par exemple, `${FilePath}` est remplacé dans le message de demande envoyé à l'agent avec `/monitored/cleanup.go`. Le message de demande est placé dans la file d'attente de commandes de l'agent. Le processeur de commandes détecte que la demande concerne un appel de programme et démarre le programme indiqué. Si une commande de type `antscript` est appelée, une nouvelle machine virtuelle Java est démarrée et la tâche Ant s'exécute sous la nouvelle machine virtuelle Java. Pour plus d'informations sur l'utilisation de la substitution de variable, voir [Personnalisation des tâches avec la substitution de variable](#).

Concepts associés

[«Personnalisation des tâches du moniteur de ressources MFT avec la substitution de variables», à la page 185](#)

Lorsque les conditions de déclenchement d'un moniteur de ressources actif sont satisfaites, la tâche définie est appelée. En plus d'appeler la tâche de transfert ou de commande avec le même agent de destination ou le même nom de fichier de destination à chaque fois, vous pouvez également modifier la définition de la tâche lors de l'exécution. Pour ce faire, vous devez insérer des noms de variable dans le fichier XML de définition de tâche. Lorsque le moniteur détermine que les conditions de déclenchement sont satisfaites et que la définition de tâche contient des noms de variable, il remplace les noms de variable par les valeurs de variable, puis appelle la tâche.

Référence associée

[«Format de message de demande de transfert de fichier», à la page 804](#)

Les transferts de fichiers sont initiés par les messages XML arrivant dans une file d'attente de commandes d'agent, généralement à la suite de l'émission d'une commande de transfert de fichiers par un utilisateur ou à l'aide de IBM MQ Explorer. Le XML de la demande de transfert doit être conforme au schéma `FileTransfer.xsd` et avoir l'élément `<request>` comme élément racine. Le document de schéma `FileTransfer.xsd` se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/schema`. Le `FileTransfer.xsd` schéma importe `fteutils.xsd`, qui se trouve dans le même répertoire.

[«Propriété `commandPath` MFT», à la page 374](#)

La propriété `commandPath` permet de spécifier les emplacements à partir desquels Managed File Transfer peut exécuter des commandes. Soyez extrêmement prudent lorsque vous définissez cette propriété car toute commande de l'un des emplacements `commandPaths` spécifiés peut être effectivement appelée à partir d'un système client distant qui peut envoyer des commandes à l'agent.

Création d'un document de définition de tâche en modifiant un document généré

Pourquoi et quand exécuter cette tâche

Vous pouvez créer le document de définition de tâche de surveillance en modifiant le document XML généré par l'option **-gt** de **fteCreateTransfer**. Le document généré comporte un élément `<request>` suivi de l'élément `<managedTransfer>`. Pour convertir cette définition de tâche en une structure `<managedCall>` valide, procédez comme suit:

Procédure

1. Remplacez les balises de début et de fin `<managedTransfer>` par des balises `<managedCall>`.
2. Supprimez tout élément `<schedule>` et noeud enfant.

3. Remplacez les balises de début et de fin <sourceAgent> par <agent> pour qu'elles correspondent aux détails de configuration de l'agent de surveillance.
4. Supprimez les éléments <destinationAgent> et <trigger> .
5. Supprimez les éléments <item> .
6. Insérez une nouvelle structure <call> . . . </call> dans l'élément <transferSet> . Cette structure contient la définition de la commande, comme illustré dans l'exemple suivant:

```
<call>
  <command name="RunCleanup.xml" type="antscript" retryCount="2"
    retryWait="30" successRC="0">
    <target>check_exists</target>
    <target>copy_to_archive</target>
    <target>rename_temps</target>
    <target>delete_files</target>
    <property name="trigger.filename" value="{FileName}"/>
    <property name="trigger.path" value="{FilePath}"/>
  </command>
</call>
```

Exemple

Vous pouvez également conserver l'élément <managedTransfer> , y compris tous les détails de transfert de fichier, et insérer jusqu'à quatre appels de commande. Dans ce cas, vous insérez une sélection des éléments d'appel suivants entre les éléments <metaDataSet> et <item> :

Appel preSource

Appelez un programme sur l'agent source avant de démarrer le transfert.

Appel postSource

Appelez un programme sur l'agent source après avoir terminé le transfert.

Appel preDestination

Appelez un programme sur l'agent de destination avant de démarrer le transfert.

Appel postDestination

Appelez un programme sur l'agent de destination une fois le transfert terminé.

Chacun de ces éléments prend la structure d'élément <command> comme décrit dans l'exemple précédent. Le schéma FileTransfer.xsd définit les types utilisés par les différents éléments d'appel.

L'exemple suivant illustre l'appel preSource, l'appel postSource, l'appel preDestination et l'appel postDestination dans un document de définition de tâche:

```
:
<transferSet priority="1">
  <metaDataSet>
    <metaData key="key1">value1</metaData>
  </metaDataSet>
  <preSourceCall>
    <command name="send.exe" retryCount="0" retryWait="0" successRC="0"
      type="executable">
      <argument>report1.pdf</argument>
      <argument>true</argument>
    </command>
  </preSourceCall>
  <postSourceCall>
    <command name="//DO_IT.JCL" retryCount="0" retryWait="0" successRC="0"
      type="jcl">
      <argument>argument</argument>
    </command>
  </postSourceCall>
  <preDestinationCall>
    <command name="ant_script.xml" retryCount="0" retryWait="0" successRC="0"
      type="antscript">
      <target>step1</target>
      <property name="name" value="value"/>
    </command>
  </preDestinationCall>
  <postDestinationCall>
    <command name="runit.cmd" retryCount="0" retryWait="0" successRC="0"/>
  </postDestinationCall>
```

```
<item checksumMethod="none" mode="binary">
```

Vous pouvez mélanger différents types de commande dans le transfert. Les éléments d'argument, de cible et de propriété sont facultatifs.

Surveillance d'un répertoire et utilisation de la substitution de variable

Vous pouvez surveiller un répertoire à l'aide de la commande **fteCreateMonitor**. La valeur d'une variable de substitution peut être remplacée dans la définition XML de la tâche et utilisée pour définir le comportement de transfert.

Pourquoi et quand exécuter cette tâche

Dans cet exemple, l'agent source est appelé AGENT_HOP. Le répertoire surveillé par AGENT_HOP est appelé /test/monitored. L'agent interroge le répertoire toutes les 5 minutes.

Une fois qu'un fichier .zip est écrit dans le répertoire, l'application qui écrit le fichier dans le répertoire écrit un fichier de déclenchement dans le même répertoire. Le nom du fichier de déclenchement est identique à celui du fichier .zip, mais il possède une extension de fichier différente. Par exemple, une fois le fichier file1.zip écrit dans le répertoire, le fichier file1.go est écrit dans le répertoire. Le moniteur de ressources surveille le répertoire pour les fichiers qui correspondent au modèle *.go, puis utilise la substitution de variable pour demander un transfert du fichier .zip associé.

Procédure

1. Créez le code XML de tâche qui définit la tâche que le moniteur exécute lorsqu'il est déclenché.

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>blue.example.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_HOP" QMgr="QM_HOP"/>
    <destinationAgent agent="AGENT_SKIP" QMgr="QM_SKIP"/>
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <file>/test/monitored/${fileName}{token=1}{separator=.}.zip</file>
        </source>
        <destination type="file" exist="overwrite">
          <file>/out/${fileName}{token=1}{separator=.}.zip</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Les variables qui sont remplacées par les valeurs associées au fichier de déclenchement sont mises en évidence en **gras**. Cette tâche XML est sauvegardée dans le fichier /home/USER1/task.xml

2. Créez un moniteur de ressources pour surveiller le répertoire /test/monitored.

Soumettez la commande suivante :

```
fteCreateMonitor -ma AGENT_HOP -mm QM_HOP -md /test/monitored
                 -mn myMonitor -mt /home/USER1/task.xml
                 -tr match,*.go -pi 5 -pu minutes
```

3. Un utilisateur ou un programme écrit le fichier jump.zip dans le répertoire /test/monitored, puis écrit le fichier jump.go dans le répertoire.
4. Le moniteur est déclenché par l'existence du fichier jump.go. L'agent substitue les informations relatives au fichier de déclenchement dans le code XML de la tâche.

Cela entraîne la transformation du code XML de la tâche en:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>blue.example.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_HOP" QMgr="QM_HOP"/>
    <destinationAgent agent="AGENT_SKIP" QMgr="QM_SKIP"/>
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <file>/test/monitored/jump.zip</file>
        </source>
        <destination type="file" exist="overwrite">
          <file>/out/jump.zip</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Résultats

Le transfert défini par la tâche XML est effectué. Le fichier `jump.zip` est lu dans le répertoire `/test/monitored` par `AGENT_HOP` et transféré dans un fichier appelé `/out/jump.zip` situé sur le système où `AGENT_SKIP` est en cours d'exécution.

Concepts associés

«Personnalisation des tâches du moniteur de ressources MFT avec la substitution de variables», à la page 185

Lorsque les conditions de déclenchement d'un moniteur de ressources actif sont satisfaites, la tâche définie est appelée. En plus d'appeler la tâche de transfert ou de commande avec le même agent de destination ou le même nom de fichier de destination à chaque fois, vous pouvez également modifier la définition de la tâche lors de l'exécution. Pour ce faire, vous devez insérer des noms de variable dans le fichier XML de définition de tâche. Lorsque le moniteur détermine que les conditions de déclenchement sont satisfaites et que la définition de tâche contient des noms de variable, il remplace les noms de variable par les valeurs de variable, puis appelle la tâche.

Tâches associées

«Configuration des tâches de surveillance MFT pour lancer des commandes et des scripts», à la page 178

Les moniteurs de ressources ne sont pas limités à l'exécution de transferts de fichiers en tant que tâche associée. Vous pouvez également configurer le moniteur pour qu'il appelle d'autres commandes à partir de l'agent de surveillance, notamment des programmes exécutables, des scripts Ant ou des travaux JCL. Pour appeler des commandes, éditez le fichier XML de définition de tâche de contrôle pour inclure un ou plusieurs éléments de commande avec les paramètres d'appel de commande correspondants, tels que les arguments et les propriétés.

Référence associée

«fteCreateMonitor: créer un moniteur de ressources MFT», à la page 424

La commande **fteCreateMonitor** crée et démarre un nouveau moniteur de ressources à partir de la ligne de commande. Vous pouvez surveiller une ressource (par exemple, le contenu d'un répertoire) à l'aide de Managed File Transfer de sorte que lorsqu'une condition de déclenchement est remplie, une tâche spécifiée, telle qu'un transfert de fichier, est démarrée.

Exemple: configuration d'une ressource MFT

Vous pouvez spécifier une file d'attente IBM MQ comme ressource à surveiller par un moniteur de ressources à l'aide du paramètre **-mq** avec la commande **fteCreateMonitor**.

Pourquoi et quand exécuter cette tâche

Dans cet exemple, la ressource à surveiller est la file d'attente *MONITORED_QUEUE*. Cette file d'attente doit se trouver sur le gestionnaire de files d'attente de l'agent de surveillance, *QM_NEPTUNE*. La condition pour laquelle la file d'attente est surveillée est la présence d'un groupe complet de messages. La tâche à exécuter si la condition est satisfaite est définie dans le fichier *task.xml*.

Remarque : Ne créez pas plus d'un moniteur de ressources pour surveiller une file d'attente individuelle. Si vous le faites, un comportement imprévisible se produit.

Procédure

Entrez la commande suivante :

```
fteCreateMonitor -ma AGENT_NEPTUNE -mn myMonitor -mm QM_NEPTUNE -mq MONITORED_QUEUE  
-mt task.xml -tr completeGroups -pi 5 -pu minutes
```

Le moniteur vérifie la file d'attente toutes les cinq minutes pour voir si la condition *completeGroups* est vraie. S'il existe un ou plusieurs groupes complets dans la file d'attente, le moniteur exécute la tâche définie dans le fichier *task.xml* une fois pour chaque groupe complet.

Personnalisation des tâches du moniteur de ressources MFT avec la substitution de variables

Lorsque les conditions de déclenchement d'un moniteur de ressources actif sont satisfaites, la tâche définie est appelée. En plus d'appeler la tâche de transfert ou de commande avec le même agent de destination ou le même nom de fichier de destination à chaque fois, vous pouvez également modifier la définition de la tâche lors de l'exécution. Pour ce faire, vous devez insérer des noms de variable dans le fichier XML de définition de tâche. Lorsque le moniteur détermine que les conditions de déclenchement sont satisfaites et que la définition de tâche contient des noms de variable, il remplace les noms de variable par les valeurs de variable, puis appelle la tâche.



Avertissement : Les noms de variable ne sont pas sensibles à la casse.

Les variables utilisées pour la substitution ne sont disponibles que pour les conditions de déclenchement positives. Seules les conditions de déclenchement *match* et *fileSize* remplacent les variables. Si une condition *noMatch* est utilisée et qu'il existe des noms de variable de substitution dans la définition de tâche, la tâche n'est pas appelée et le moniteur émet le code retour 110 et le message d'erreur BFGDM0060E.

Si la ressource contrôlée est une file d'attente

La valeur de toute propriété de message IBM MQ du premier message à lire dans la file d'attente surveillée peut être remplacée dans la définition XML de la tâche.

Les propriétés de message définies par l'utilisateur sont préfixées avec *usr.* mais n'incluent pas ce préfixe dans le nom de la variable. Les noms de variable doivent être précédés d'un signe dollar (\$) et placés entre accolades {}.

Par exemple, `${destFileName}` est remplacé par la valeur de la propriété de message `usr.destFileName` du premier message à lire dans la file d'attente source. Pour plus d'informations, voir «[Propriétés de message IBM MQ lues par MFT à partir des messages des files d'attente source](#)», à la page 701 et «[Surveillance d'une file d'attente et utilisation de la substitution de variable](#)», à la page 191.

Si une variable n'est pas définie en tant que propriété de message, le moniteur signale une erreur BFGDM0060E et renvoie le code retour 110 (échec de la substitution de la variable de la tâche de surveillance). En outre, l'agent écrit le message d'erreur suivant dans son journal des événements (`outputN.log`):

```
BFGDM0113W: Trigger failure for <monitor name> for reason BFGDM0060E: A monitor task could  
not complete as a variable substitution <variable name> was not present.
```

Si la consignation du moniteur de ressources modéré ou prolix est activée pour le moniteur, le moniteur écrit le message suivant dans le journal des événements du moniteur de ressources de l'agent (resmoneventN.log):

```
BFGDM0060E: A monitor task could not complete as a variable substitution
<variable name> was not present.
```

Pour plus d'informations sur la journalisation du moniteur de ressources, voir «[Consignation des moniteurs de ressources MFT](#)», à la page 196 .

Le tableau suivant indique les variables de substitution fournies par défaut. Par exemple, `${AGENTNAME}` est remplacé par le nom de l'agent du moniteur de ressources.

Tableau 18. Variables de substitution fournies par défaut

Variable	Description
AgentName	Nom de l'agent du moniteur de ressources.
QUEUENAME	Nom de la file d'attente surveillée.
ENCODING	Codage de caractères du premier message de la file d'attente ou du premier message d'un groupe.
ID message	ID message IBM MQ du premier message de la file d'attente ou du premier message du groupe.
groupID	ID groupe IBM MQ du groupe ou ID message si un seul message est trouvé. Cette variable est uniquement définie si vous contrôlez des groupes complets.
CurrentTimestamp	Horodatage basé sur l'heure locale à laquelle le moniteur a été déclenché. La valeur d'horodatage est unique pour l'agent.
CurrentTimeStamp UTC	Horodatage basé sur l'heure, dans le fuseau horaire UTC, à laquelle le moniteur a été déclenché. La valeur d'horodatage est unique pour l'agent.

Si la ressource contrôlée est un répertoire

Le tableau suivant présente l'ensemble des noms de variable pouvant être remplacés dans la définition XML de la tâche.

Tableau 19. Variables pouvant être substituées

Variable	Description
FilePath	Nom de chemin complet du fichier de déclenchement.
FileName	Partie du nom de fichier du déclencheur.
Heure de LastModified	Heure de la dernière modification du fichier de déclenchement. Cette heure est exprimée en tant qu'heure locale du fuseau horaire dans lequel l'agent s'exécute et est formatée en tant qu'heure ISO 8601.
Date de LastModified	Date de la dernière modification du fichier de déclenchement. Cette date est exprimée en tant que date locale du fuseau horaire dans lequel l'agent s'exécute et est formatée en tant que date ISO 8601.
LastModifiedTimeUTC	Heure de la dernière modification du fichier de déclenchement. Cette heure est exprimée sous la forme de l'heure locale convertie au fuseau horaire UTC et est formatée sous la forme d'une heure ISO 8601
LastModifiedDateUTC	Date de la dernière modification du fichier de déclenchement. Cette date est exprimée sous forme de date locale convertie au fuseau horaire TUC et est formatée sous forme de date ISO 8601.

<i>Tableau 19. Variables pouvant être substituées (suite)</i>	
Variable	Description
AgentName	Nom de l'agent du moniteur de ressources.
CurrentTimeSt amp	Horodatage basé sur l'heure locale à laquelle le moniteur a été déclenché. La valeur d'horodatage est unique pour l'agent.
CurrentTimeSt ampUTC	Horodatage basé sur l'heure dans le fuseau horaire UTC à laquelle le moniteur a été déclenché. La valeur d'horodatage est unique pour l'agent.

Si la ressource contrôlée est un fichier de déclenchement

Le tableau suivant présente l'ensemble des noms de variable pouvant être remplacés lorsqu'un moniteur de ressources utilise le contenu d'un fichier de déclenchement pour déterminer les fichiers à transférer.

<i>Tableau 20. Variables pouvant être remplacées lors de l'utilisation d'un fichier de déclenchement</i>	
Variable	Description
contentSource	Nom de chemin complet du fichier source.
contentDestina tion	Nom de chemin complet du fichier de destination.

Les noms de variable doivent être précédés d'un signe dollar (\$) et placés entre accolades, { }. Par exemple, `${FilePath}` est remplacé par le chemin de fichier qualifié complet du fichier de déclenchement correspondant.

Il existe deux mots clés spéciaux qui peuvent être appliqués aux noms de variable pour fournir des améliorations supplémentaires. Il s'agit des fonctions suivantes :

sème

Index de jeton à remplacer (à partir de 1 à partir de la gauche et à partir de -1 à partir de la droite)

séparateur

Caractère unique permettant de segmenter la valeur de la variable. La valeur par défaut est la barre oblique (/) sur les plateformes UNIX ou la barre oblique inversée (\) sur les plateformes Windows , mais le séparateur peut être tout caractère valide pouvant apparaître dans la valeur de la variable.

Si le mot clé de séparation est indiqué dans un nom de variable, la valeur de la variable est divisée en unités lexicales en fonction du caractère de séparation.

La valeur affectée au mot clé token est utilisée comme index pour sélectionner le jeton à utiliser pour remplacer le nom de la variable. L'index de jeton est relatif au premier caractère de la variable et commence à 1. Si le mot clé token n'est pas spécifié, la totalité de la variable est insérée.

Toutes les valeurs qui sont substituées dans un nom d'agent dans le message XML sont traitées de manière non sensible à la casse. Tous les noms Managed File Transfer Agent sont en majuscules. Si la valeur Paris est remplacée par un attribut d'agent dans le message XML, cette valeur est interprétée comme une référence à l'agent PARIS.

Concepts associés

«Exemples: substitution de variable pour les définitions de moniteur de ressources», à la page 188
Exemples de substitution de variables pour les définitions de moniteur de ressources à l'aide de XML et de IBM MQ Explorer.

Référence associée

«Que faire si la substitution de variables entraîne l'accès de plusieurs fichiers à un même nom de fichier», à la page 337

Pour Managed File Transfer, si vous surveillez un répertoire et transférez plusieurs fichiers d'une source vers un emplacement de destination et que vous utilisez la substitution de variable `${FileName}` , vous devez tester les résultats de la substitution de variable. Les résultats doivent être testés car l'utilisation

de la substitution de variables peut entraîner l'appel de combinaisons inattendues de commandes de transfert de fichiers.

Exemples: substitution de variable pour les définitions de moniteur de ressources

Exemples de substitution de variables pour les définitions de moniteur de ressources à l'aide de XML et de IBM MQ Explorer.

Exemples illustrant le fonctionnement de la substitution de variable

En supposant que le chemin d'accès au fichier de déclenchement correspondant est `c:\MONITOR\REPORTS\Paris\Report2009.doc` sur les plateformes Windows et `/MONITOR/REPORTS/Paris/Report2009.doc` sur les plateformes UNIX, les variables sont remplacées comme indiqué dans le tableau suivant.

Tableau 21. Mode de substitution des variables

Spécification de variable	Après la substitution de variable
<code>\${FilePath}</code>	Windows : <code>c:\MONITOR\REPORTS\Paris\Report2009.doc</code> UNIX : <code>/MONITOR/REPORTS/Paris/Report2009.doc</code>
<code>\${FilePath{token=1}{separator=.}}</code>	Windows : <code>c:\MONITOR\REPORTS\Paris\Report2009</code> UNIX : <code>/MONITOR/REPORTS/Paris/Report2009</code>
<code>\${FilePath{token=2}{separator=.}}</code>	Windows : <code>doc</code> UNIX : <code>doc</code>
<code>\${FilePath{token=3}}</code>	Windows : <code>RAPPORTS</code> UNIX : <code>Paris</code>

Vous pouvez également spécifier un index de jeton négatif pour sélectionner des jetons par rapport au dernier caractère de la variable, comme indiqué dans le tableau suivant. Les exemples du tableau utilisent la même valeur de variable, `c:\MONITOR\REPORTS\Paris\Report2009.doc` sous Windows et `/MONITOR/REPORTS/Paris/Report2009.doc` sous UNIX.

Tableau 22. Exemples d'utilisation d'un index de jeton négatif

Spécification de variable	Après la substitution de variable
<code>\${FilePath}</code>	Windows : <code>c:\MONITOR\REPORTS\Paris\Report2009.doc</code> UNIX : <code>/MONITOR/REPORTS/Paris/Report2009.doc</code>
<code>\${FilePath{token=-2}{separator=.}}</code>	Windows : <code>c:\MONITOR\REPORTS\Paris\Report2009</code> UNIX : <code>/MONITOR/REPORTS/Paris/Report2009</code>
<code>\${FilePath{token=-2}{separator=\\}}</code>	Windows : <code>Paris</code> UNIX : <code>Paris</code>

Tableau 22. Exemples d'utilisation d'un index de jeton négatif (suite)

Spécification de variable	Après la substitution de variable
<code>\${FilePath{token=-4}}</code>	Windows : Monitor UNIX : Monitor

Les variables utilisées pour la substitution ne sont disponibles que pour les conditions de déclenchement positives suivantes et l'option `noSizeChange`, qui est une exception à la règle de condition de déclenchement positive:

- `match`
- `fileSize`
- `noSize-Modification`

Si une condition `noMatch` est utilisée et qu'il existe des noms de variable de substitution dans la définition de tâche, la tâche n'est pas appelée et le moniteur émet le code retour 110 et le message d'erreur `BFGDM0060E`.

Exemple d'utilisation de XML

L'exemple suivant de définition de tâche XML utilise le nom de l'agent de surveillance comme agent source pour le transfert (Paris), utilise l'avant-dernier nom de répertoire dans le chemin de fichier comme nom de l'agent de destination pour le transfert (Report2009) et renomme le fichier transféré pour qu'il soit la racine du nom du fichier de déclenchement avec l'extension `.rpt`.

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="${AgentName}" QMgr="QM1"/>
    <destinationAgent agent="${FilePath{token=-2}}" QMgr="QMD"/>
    <transferSet>
      <item mode="binary" checksumMethod="MD5">
        <source recursive="false" disposition="leave">
          <file>c:/incoming/reports/summary/report.doc</file>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/${FileName{token=1}{separator=.}}.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Cela entraîne la transformation du code XML de la tâche en:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT1" QMgr="QM1"/>
    <destinationAgent agent="Paris" QMgr="QMD"/>
    <transferSet>
      <item mode="binary" checksumMethod="MD5">
        <source recursive="false" disposition="leave">
          <file>c:/incoming/reports/summary/report.doc</file>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/Report2009.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

```
</transferSet>
</managedTransfer>
</request>
```

La variable `${FilePath}{token=-2}` dans l'attribut agent de l'élément `<destinationAgent>` est remplacée par la valeur `Paris`. Cette valeur est traitée de manière non sensible à la casse et interprétée comme une référence à l'agent `PARIS`.

Exemples d'utilisation de IBM MQ Explorer

Lors de la création d'un moniteur de ressources via IBM MQ Explorer, et une fois que les propriétés du moniteur et les conditions de déclenchement ont été spécifiées, l'option permet d'ajouter des éléments de transfert au moniteur. Les exemples suivants montrent comment les variables `${FilePath}` et `${FileName}` peuvent être utilisées dans le "**panneau Ajouter un élément de transfert**" pour personnaliser les transferts résultant d'une correspondance de moniteur de ressources.

Exemple 1

Afin de simplement transférer le fichier source à un autre emplacement lorsqu'une condition de déclenchement est remplie, la variable `${FilePath}` peut être utilisée:

- Définissez le **nom de fichier** source sur `${FilePath}`.
- Dans le menu déroulant **Type** de la destination, sélectionnez **Annuaire**.
- Définissez le **nom de fichier** de destination comme emplacement vers lequel vous souhaitez que le fichier source soit transféré. Par exemple, il peut s'agir de `C:\MFT\out\`.

Exemple 2

Pour transférer le fichier source vers un autre emplacement et modifier l'extension du fichier, la variable `${FileName}` peut être utilisée avec la variable `${FilePath}` :

Dans l'exemple suivant, on suppose que le chemin du fichier source est égal à `C:\MONITOR\REPORTS\Paris\Report2009.doc`:

- Définissez le **nom de fichier** source sur `${FilePath}`.
- Définissez le **nom de fichier** de destination comme emplacement vers lequel vous souhaitez que le fichier source soit transféré, suivi de `${FileName}{token=1}{separator=.}`, suivi de la nouvelle extension du fichier. Par exemple, il peut s'agir de `C:\MFT\out\${FileName}{token=1}{separator=.}.rpt`, qui correspond à `C:\MFT\out\Report2009.rpt` avec le nom du fichier source.

Exemple 3

Afin d'utiliser une partie du chemin du fichier source pour déterminer la destination du transfert, la variable `${FilePath}` peut être utilisée avec des spécifications de jeton et de séparateur.

Dans l'exemple suivant, on suppose que le chemin du fichier source est égal à `C:\MONITOR\REPORTS\Paris\Report2009.doc`.

Il est possible d'utiliser une partie du chemin du fichier source pour déterminer la destination du fichier. A l'aide de l'exemple de chemin de fichier `C:\MONITOR\REPORTS\Paris\Report2009.doc`, si le fichier devait être transféré dans un dossier en fonction de l'emplacement du fichier source, c'est-à-dire `Paris` dans cet exemple, les opérations suivantes peuvent être effectuées:

- Définissez le **nom de fichier** source sur `${FilePath}`.
- Définissez le **Nom de fichier** de destination comme destination dans laquelle se trouvent les dossiers de chaque emplacement, puis ajoutez la partie de destination du chemin de fichier et le nom de fichier. Par exemple, il peut s'agir de `C:\MFT\out\${FilePath}{token=-2}{separator=\}\${FileName}`, qui correspond à `C:\MFT\out\Paris\Report2009.doc` avec le nom du fichier source.

Concepts associés

«Personnalisation des tâches du moniteur de ressources MFT avec la substitution de variables», à la page 185

Lorsque les conditions de déclenchement d'un moniteur de ressources actif sont satisfaites, la tâche définie est appelée. En plus d'appeler la tâche de transfert ou de commande avec le même agent de destination ou le même nom de fichier de destination à chaque fois, vous pouvez également modifier la définition de la tâche lors de l'exécution. Pour ce faire, vous devez insérer des noms de variable dans le fichier XML de définition de tâche. Lorsque le moniteur détermine que les conditions de déclenchement sont satisfaites et que la définition de tâche contient des noms de variable, il remplace les noms de variable par les valeurs de variable, puis appelle la tâche.

Référence associée

«Que faire si la substitution de variables entraîne l'accès de plusieurs fichiers à un même nom de fichier», à la page 337

Pour Managed File Transfer, si vous surveillez un répertoire et transférez plusieurs fichiers d'une source vers un emplacement de destination et que vous utilisez la substitution de variable `${FileName}`, vous devez tester les résultats de la substitution de variable. Les résultats doivent être testés car l'utilisation de la substitution de variables peut entraîner l'appel de combinaisons inattendues de commandes de transfert de fichiers.

Surveillance d'une file d'attente et utilisation de la substitution de variable

Vous pouvez surveiller une file d'attente et transférer des messages de la file d'attente surveillée vers un fichier à l'aide de la commande **fteCreateMonitor**. La valeur de toute propriété de message IBM MQ dans le premier message à lire dans la file d'attente surveillée peut être remplacée dans la définition XML de tâche et utilisée pour définir le comportement de transfert.

Pourquoi et quand exécuter cette tâche

Dans cet exemple, l'agent source est appelé AGENT_VENUS, qui se connecte à QM_VENUS. La file d'attente surveillée par AGENT_VENUS est appelée START_QUEUE et se trouve sur QM_VENUS. L'agent interroge la file d'attente toutes les 30 minutes.

Lorsqu'un groupe complet de messages est écrit dans la file d'attente, la tâche de surveillance envoie le groupe de messages dans un fichier à l'un des agents de destination, qui se connectent tous au gestionnaire de files d'attente QM_MARS. Le nom du fichier vers lequel le groupe de messages est transféré est défini par la IBM MQ propriété de message `usr.fileName` sur le premier message du groupe. Le nom de l'agent auquel le groupe de messages est envoyé est défini par la IBM MQ propriété de message `usr.toAgent` sur le premier message du groupe. Si l'en-tête `usr.toAgent` n'est pas défini, la valeur par défaut à utiliser pour l'agent de destination est AGENT_MAGENTA.

Lorsque vous spécifiez `useGroups="true"`, si vous ne spécifiez pas également `groupId="${GROUPID}"`, le transfert prend simplement le premier message de la file d'attente. Par exemple, si vous utilisez la substitution de variable pour générer le `fileName`, il est possible que le contenu de `a.txt` ne soit pas correct. En effet, `fileName` est généré par le moniteur, mais le transfert reçoit en fait un message qui n'est pas celui qui doit générer le fichier appelé `fileName`.

Procédure

1. Créez le code XML de tâche qui définit la tâche que le moniteur exécute lorsqu'il est déclenché.

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:namespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_VENUS" QMgr="QM_VENUS"/>
    <destinationAgent agent="${toAgent}" QMgr="QM_MARS"/>
  </transferSet>
</request>
```

```

<item mode="binary" checksumMethod="none">
  <source>
    <queue useGroups="true" groupId="{GROUPEID}">START_QUEUE</queue>
  </source>
  <destination type="file" exist="overwrite">
    <file>/reports/{fileName}.rpt</file>
  </destination>
</item>
</transferSet>
</managedTransfer>
</request>

```

Les variables qui sont remplacées par les valeurs des en-têtes de message IBM MQ sont mises en évidence en **gras**. Cette tâche XML est sauvegardée dans le fichier /home/USER1/task.xml

2. Créez un moniteur de ressources pour surveiller la file d'attente START_QUEUE.

Soumettez la commande suivante :

```

fteCreateMonitor -ma AGENT_VENUS -mm QM_VENUS -mq START_QUEUE
                 -mn myMonitor -mt /home/USER1/task.xml
                 -tr completeGroups -pi 30 -pu minutes -dv toAgent=AGENT_MAGENTA

```

3. Un utilisateur ou un programme écrit un groupe de messages dans la file d'attente START_QUEUE. Les propriétés de message IBM MQ suivantes sont définies pour le premier message de ce groupe:

```

usr.fileName=larmer
usr.toAgent=AGENT_VIOLET

```

4. Le moniteur est déclenché lorsque le groupe complet est écrit. L'agent remplace les propriétés de message IBM MQ dans le code XML de la tâche.

Cela entraîne la transformation du code XML de la tâche en:

```

<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_VENUS" QMgr="QM_VENUS"/>
    <destinationAgent agent="AGENT_VIOLET" QMgr="QM_MARS"/>
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <queue useGroups="true" groupId="{GROUPEID}">START_QUEUE</queue>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/larmer.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>

```

Résultats

Le transfert défini par la tâche XML est effectué. Le groupe complet de messages lus à partir de la file d'attente START_QUEUE par AGENT_VENUS est écrit dans un fichier appelé /reports/larmer.rpt sur le système où s'exécute AGENT_VIOLET.

Que faire ensuite

Transfert de chaque message dans un fichier distinct

Si vous souhaitez surveiller une file d'attente et que chaque message soit transféré dans un fichier distinct, vous pouvez utiliser une technique similaire à celle décrite précédemment dans cette rubrique.

1. Créez le moniteur comme décrit précédemment, en spécifiant le paramètre **-tr completeGroups** dans la commande **fteCreateMonitor**.
2. Dans le code XML de la tâche, indiquez ce qui suit:

```
<queue useGroups="true" groupId="${GROUPID}">START_QUEUE</queue>
```

Toutefois, lorsque vous placez les messages dans la file d'attente source, ne les placez pas dans un groupe IBM MQ. Ajoutez des propriétés de message IBM MQ à chaque message. Par exemple, spécifiez la propriété `usr.filename` avec une valeur de nom de fichier unique pour chaque message. Ainsi, Managed File Transfer Agent traite chaque message de la file d'attente source comme un groupe distinct.

Surveiller le comportement des nouvelles tentatives pour les transferts de messages vers des fichiers

Si un transfert de message à fichier déclenché par un moniteur de ressources échoue et quitte le groupe de messages qui a déclenché le moniteur dans la file d'attente, ce transfert est soumis à nouveau à des intervalles d'interrogation ultérieurs. Le nombre de fois où le transfert est soumis à nouveau est limité par la propriété **monitorGroupRetryLimit** de l'agent de surveillance.

Le nombre de fois où le transfert de message à fichier a été déclenché est déterminé à partir du nombre d'annulations MQMD du premier message du groupe.

Chaque fois qu'un nouveau transfert de message à fichier est déclenché, un nouvel ID de transfert est généré pour la tâche de transfert.

Si l'agent est redémarré, le moniteur déclenche à nouveau un transfert même si le nombre de fois où le transfert a été déclenché a dépassé la valeur de **monitorGroupRetryLimit**. Si cette tentative de transfert entraîne le nombre de fois où le transfert a été déclenché pour dépasser la valeur de **monitorGroupRetryLimit**, l'agent écrit une erreur dans son journal des événements.

Un message unique est traité comme s'il s'agissait d'un groupe unique, et le transfert est déclenché à nouveau à chaque intervalle d'interrogation alors que le message reste dans la file d'attente et que le nombre de fois où le transfert a été déclenché est inférieur à la valeur de **monitorGroupRetryLimit**.

Définition de la propriété **monitorGroupRetryLimit**

La valeur de la propriété **monitorGroupRetryLimit** correspond au nombre maximal de fois où un moniteur déclenche à nouveau un transfert de message à fichier si le groupe de messages existe toujours dans la file d'attente. La valeur par défaut de cette propriété est 10. La valeur de cette propriété peut être définie sur n'importe quelle valeur entière positive ou sur -1. Si la valeur -1 est spécifiée pour cette propriété, le moniteur déclenche à nouveau le transfert un nombre illimité de fois, jusqu'à ce que la condition de déclenchement ne soit pas satisfaite.

Pour définir la propriété **monitorGroupRetryLimit** sur l'agent de surveillance, procédez comme suit:

1. Arrêtez l'agent de surveillance à l'aide de la commande **fteStopAgent**.
2. Editez le fichier `agent.properties` de l'agent de surveillance pour inclure la ligne `monitorGroupRetryLimit=number_of_retries`. Le fichier `agent.properties` se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/monitoring_agent_name`.
3. Démarrez l'agent de surveillance à l'aide de la commande **fteStartAgent**.

Tâches associées

«Exemple: configuration d'une ressource MFT», à la page 184

Vous pouvez spécifier une file d'attente IBM MQ comme ressource à surveiller par un moniteur de ressources à l'aide du paramètre **-mq** avec la commande **fteCreateMonitor**.

Référence associée

«Le fichier MFT `agent.properties`», à la page 564

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.

Utilisation d'un fichier de déclenchement

Vous pouvez utiliser le contenu d'un fichier de déclenchement dans un moniteur de ressources pour définir un ensemble de fichiers à transférer dans une demande de transfert unique. Chaque fois qu'un fichier de déclenchement correspondant est détecté, son contenu est analysé pour les chemins d'accès aux fichiers source et éventuellement pour les chemins d'accès aux fichiers de destination. Ces chemins de fichier sont ensuite utilisés pour définir des éléments de fichier dans le fichier XML de transfert de tâche que vous spécifiez, qui est soumis en tant que demande de transfert unique à l'agent. La définition du moniteur de ressources détermine si le contenu du déclencheur est activé.

Vous pouvez activer le déclenchement de contenu de fichier lorsque vous créez un moniteur en spécifiant le paramètre **-tc** (contenu de déclencheur). Ce paramètre **-tc** s'applique uniquement aux options de déclencheur de fichier `match` et `noSizeChange`. Pour plus d'informations sur la création d'un moniteur, voir «[fteCreateMonitor: créer un moniteur de ressources MFT](#)», à la page 424.

Lors de l'utilisation d'un fichier de contenu de déclencheur, le format par défaut de chaque ligne est:

- Un chemin de fichier source unique, ou
- Un chemin d'accès au fichier source et un chemin d'accès au fichier de destination, séparés par une virgule

où les espaces blancs sont traités dans le cadre des chemins d'accès aux fichiers. Il est possible de modifier le format de ligne par défaut en spécifiant les paramètres **-tcr** et **-tcc** dans la commande **fteCreateMonitor**. Pour plus d'informations, voir «[Options avancées](#)», à la page 195.

Une fois qu'un fichier de déclenchement est analysé, une liste de chemins de fichier est générée et appliquée au fichier XML de la tâche de transfert que vous avez spécifié. Comme pour tous les moniteurs, le format XML de la tâche de transfert est un XML de tâche de transfert complet généré par la commande **fteCreateTransfer** avec un seul élément ou fichier défini. L'élément unique doit utiliser les variables de substitution `{contentSource}`, et éventuellement `{contentDestination}`, comme remplacements pour les chemins de fichier source et de destination. Le moniteur développe le fichier XML de la tâche de transfert pour inclure un élément de fichier pour chaque ligne (chemin de fichier) dans le fichier de déclenchement.

Vous ne pouvez pas utiliser le déclenchement de contenu de fichier avec le paramètre **-bs** car le paramètre **-tc** implique une demande de transfert pour chaque fichier de déclenchement.

Exemple

L'exemple suivant définit un moniteur à déclencher sur un fichier qui se termine par `trig` et lit les chemins d'accès à ce fichier.

```
fteCreateTransfer -gt task.xml -sa SrcAgent -da DestAgent -dd /file/destdir ${contentSource}
fteCreateMonitor -mn TrigMonitor -md /home/trigdir -mt task.xml -ma SrcAgent -tr "match,*.trig"
-tc
```

La commande **fteCreateTransfer** crée un fichier appelé `task.xml` pour un fichier unique avec un chemin de fichier source de `{sourceContent}`. Exemple :

```
<item checksumMethod="MD5" mode="binary">
  <source disposition="leave" recursive="false">
    <file>${contentSource}</file>
  </source>
</item>
```

La commande **fteCreateMonitor** recherche les fichiers qui se terminent par `trig` dans le répertoire `/home/trigdir` et utilise le contenu pour créer une demande de transfert unique basée sur `task.xml`

pour tous les chemins de ce fichier de déclenchement. Le format du fichier de déclenchement doit être un chemin de fichier (source uniquement) sur chaque ligne sans séparateur de virgule. Exemple :

```
/home/file/first.txt
/home/file/second.txt
/home/different/third.txt
:
```

Tous les fichiers sont distribués dans le répertoire `/file/destdir` avec son nom de fichier et non son chemin d'accès, c'est-à-dire que `/home/file/first.txt` est distribué à `/file/destdir/first.txt`.

Sinon, si vous remplacez le paramètre `-dd /file/destdir` dans la commande **fteCreateTransfer** par `-df ${contentDestination}` et le format du contenu d'un fichier de déclenchement par *chemin du fichier source, chemin du fichier de destination*, vous pouvez définir des chemins de destination différents pour le même agent de destination. Exemple :

```
/home/file/first.txt,/home/other/sixth.txt
```

L'emplacement de destination devient alors `/home/other/sixth.txt`.

Les variables de substitution peuvent être segmentées. Par exemple, vous pouvez séparer la partie nom de fichier du chemin fourni à l'aide de `${contentDestination{token=-1}}`. Par conséquent, si la destination **fteCreateTransfer** est définie comme `-df /file/destdir/${contentDestination{token=-1}}`, la nouvelle destination pour `/home/file/first.txt` est `/file/destdir/sixth.txt`.

Options avancées

Vous pouvez modifier le format de ligne par défaut pour le contenu du fichier de déclenchement à l'aide du paramètre `-tcr regex`. Indiquez une expression régulière qui correspond au format de ligne requis et fournit un ou deux groupes de capture. Le premier groupe de capture est la source et le second, facultatif, le groupe de capture est la destination. Exemple :

- Les chemins source et de destination sont séparés par un trait d'union:

```
((?:[^-]+)-((?:[^-]+))
```

Dans cet exemple, le séparateur est défini dans trois emplacements et les trois instances du trait d'union, -, peuvent être remplacées par n'importe quel caractère. Veillez à mettre en échappement les caractères spéciaux.

- Les chemins source et de destination sont séparés par une virgule avec des espaces de fin. Les commentaires indiqués par un signe dièse (#) sont ignorés.

```
((?:[^\, ]+), ((?:[^\, ]+)) *(?:#.*)+
```

Les chemins de fichier ne peuvent pas contenir le signe dièse (#). En règle générale, une entrée se présente comme suit: `/home/source/from.txt,/home/destination/to.txt # some comment`.

Si vous utilisez le paramètre `-tcr`, assurez-vous que l'expression régulière est bien conçue et testée de sorte que l'expression puisse détecter les erreurs et analyser correctement les fichiers de déclenchement.

Vous pouvez inverser l'ordre de la capture à l'aide du paramètre `-tcc destSrc`. Si vous spécifiez ce paramètre, le premier groupe de capture est le chemin d'accès au fichier de destination et le deuxième groupe est le chemin d'accès au fichier source.

Comment les erreurs sont traitées

Fichier de déclenchement vide

Si le fichier de déclenchement est vide, le résultat n'est pas un transfert de fichier. Autrement dit, le moniteur crée une demande de transfert mais aucun élément de fichier n'est spécifié.

Fichier de déclenchement comportant des erreurs

Si une entrée d'un fichier de déclenchement ne parvient pas à être analysée par rapport au format attendu, aucune demande de transfert n'est générée. Un journal des erreurs du moniteur est publié et l'erreur est également consignée dans le journal des événements. Le fichier de déclenchement est marqué comme étant traité et le moniteur ne tente pas de le traiter à nouveau tant que le fichier n'a pas été mis à jour.

XML de tâche de transfert non concordant

Le code XML de la tâche de transfert doit correspondre au fichier de déclenchement, c'est-à-dire que si le code XML de la tâche de transfert contient à la fois `${sourceContent}` et `${destinationContent}`, tous les fichiers de déclenchement de ce moniteur doivent avoir des chemins d'accès aux fichiers source et de destination et inversement. Dans le premier cas, le moniteur signale un échec de substitution du `${destinationContent}` si le fichier de déclenchement fournit uniquement le chemin du fichier source.

Exemples

L'exemple suivant est un déclencheur de contenu de base dans lequel le contenu d'un fichier de déclencheur comporte uniquement un chemin d'accès au fichier source:

```
fteCreateTransfer -gt task.xml -sa SrcAgent -da DestAgent -dd /file/destdir ${sourceContent}
fteCreateMonitor -mn TrigMonitor -md /home/trigdir -mt task.xml -ma SrcAgent -tr "match,*.trig"
-tc
```

Le paramètre **-tcr** définit deux groupes de capture d'une séquence de caractères séparés par un espace. Le paramètre et l'option **-tcc destSrc** indiquent que les groupes de capture doivent être traités en tant que destination puis source.

```
fteCreateTransfer -gt task.xml -sa SrcAgent -da DestAgent -df ${destinationContent} $
{sourceContent}
fteCreateMonitor -mn TrigMonitor -md /home/trigdir -mt task.xml -ma SrcAgent -tr "match,*.trig"
-tc
-tcr "((?:[^\ ])+) ((?:[^\ ])+)" -tcc destSrc
```

V 9.0.3

Consignation des moniteurs de ressources MFT

Depuis la IBM MQ 9.0.3, Managed File Transfer inclut une méthode améliorée d'obtention d'informations de diagnostic sur les moniteurs de ressources.

Pourquoi et quand exécuter cette tâche

Depuis IBM MQ 9.0.3, vous pouvez utiliser la consignation pour les moniteurs de ressources. Vous pouvez utiliser la commande **fteSetAgentLogLevel** ou le fichier `agent.properties` pour contrôler la consignation du moniteur de ressources.

Notez que les points de trace existants sont toujours utilisés pour capturer des informations.

Les journaux du moniteur de ressources sont écrits dans un fichier nommé `resmoneventN.log`, où *N* représente un nombre ; par exemple, `resmonevent0.log`.



Attention : Tous les moniteurs de ressources d'un agent écrivent dans le même fichier journal.

Pour obtenir un exemple de sortie d'un fichier `resmoneventN.log`, voir [«Que faire si votre moniteur de ressources de répertoire MFT ne déclenche pas de fichiers»](#), à la page 322.

Le tableau suivant répertorie les types d'événements que le moniteur de ressources écrit dans le fichier journal. La troisième colonne décrit le niveau de journalisation nécessaire pour capturer chaque événement où le niveau le plus bas est INFO et le niveau le plus élevé est VERBOSE.

Notez que la définition d'un niveau de journalisation supérieur permet également d'écrire des événements de niveau inférieur. Par exemple, la définition du niveau de journalisation sur MODÉRE permet également d'écrire des événements de niveau INFO, mais pas des événements de niveau VERBOSE.

Numéro	Événement	Niveau de journalisation	Description
1	Créer un moniteur	INFO	Un moniteur de ressources a été créé.
2	Suppression du moniteur	INFO	Un moniteur de ressources a été supprimé.
3	Arrêt du moniteur	INFO	Un moniteur de ressources a été arrêté.
4	Démarrage du moniteur	INFO	Un moniteur de ressources a été démarré.
5	Interrogation de démarrage du moniteur	INFO	Un moniteur de ressources a démarré un nouveau cycle d'interrogation.
6	Fin de l'interrogation du moniteur	INFO	Un cycle d'interrogation du moniteur de ressources s'est arrêté.
7	Correspondance de modèle	PROLIXE	Un fichier dans le répertoire du moniteur de déclenchement ou un message dans une file d'attente correspondant au modèle spécifié a été trouvé.
8	Non-concordance de modèle	PROLIXE	Un fichier non concordant dans le répertoire du moniteur de déclenchement ou un message dans une file d'attente qui ne correspond pas au modèle spécifié a été trouvé.
9	Soumission de transfert	INFO	Un transfert a été lancé par le moniteur de ressources.
10	Répertoire trop profond	PROLIXE	Le répertoire contrôlé par le moniteur de ressources contient plus de sous-répertoires à interroger que le nombre indiqué dans la configuration du moniteur de ressources.
11	Fichier verrouillé	MODERE	Le fichier de déclenchement surveillé par le moniteur de ressources est verrouillé par un autre processus.
12	Taille de fichier réduite	MODERE	Le fichier de déclenchement est inférieur à la taille spécifiée dans la configuration du moniteur de ressources.
13	Taille de fichier instable	MODERE	Le fichier de déclenchement est modifié plus fréquemment que prévu par la configuration du moniteur de ressources.
14	Trop d'interrogations	MODERE	Un moniteur de ressources a interrogé un fichier de déclenchement instable trop de fois.
15	Éléments mis en correspondance	INFO	Nombre total de fichiers de déclenchement trouvés dans le répertoire interrogé par un moniteur de ressources.

Numéro	Événement	Niveau de journalisation	Description
16	Articles en cours de transfert	INFO	Nombre total d'articles dans la demande de transfert.
17	FDC ou exception	MODERE	Un moniteur de ressources a généré une exception.
18	Demande de transfert	INFO	Demande de transfert soumise par le moniteur de ressources.
19	Echec du démarrage du moniteur	MODERE	Le démarrage d'un moniteur de ressources a échoué.

Procédure

- Pour utiliser le **fteSetAgentLogLevel** afin d'activer et de désactiver la journalisation du moniteur de ressources, voir [fteSetAgentLogLevel](#) pour obtenir une description du paramètre **logMonitor** et des exemples d'utilisation des différentes options.
- Pour utiliser le fichier `agent.properties` afin de contrôler la consignation du moniteur de ressources, voir «Le fichier MFT `agent.properties`», à la page 564 pour une description des propriétés supplémentaires qui vous permettent d'effectuer les activités de consignation suivantes:
 - Activer ou désactiver la journalisation
 - Limiter la taille de chaque fichier journal
 - Limitation du nombre de journaux que les moniteurs de ressources peuvent générer

Exemple

Les exemples de message suivants définissent la verbose consignation de niveau pour l'agent HA2, sur le gestionnaire de files d'attente MFTDEMO:

```
<?xml version="1.0"?>
<log:log version="6.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xmlns:log="https://www.ibm.com/log">
  <log:originator>
    <log:request>
      <log:hostname>192.168.7.1</log:hostname>
      <log:userID>johndoe</log:userID>
    </log:request>
  </log:originator>
  <log:endpoint agent="HA2" QMgr="MFTDEMO"/>
  <log:logMonitor>MON1="verbose"</log:logMonitor>
</log:log>
```

Référence associée

Commande `fteSetAgentLogLevel`

La commande **fteSetAgentLogLevel** permet d'activer ou de désactiver la consignation des informations de diagnostic pour les transferts de fichiers entre un agent de pont de protocole Managed File Transfer et des serveurs de fichiers FTP/SFTP/FTPS.

«Le fichier MFT `agent.properties`», à la page 564

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.

Sauvegarde et restauration des moniteurs de ressources MFT

Vous pouvez sauvegarder les moniteurs de ressources que vous souhaitez pouvoir utiliser ultérieurement en exportant leurs définitions dans un fichier XML que vous pouvez ensuite importer pour créer un nouveau moniteur de ressources à partir de la sauvegarde.

Pourquoi et quand exécuter cette tâche

Vous devrez peut-être sauvegarder les moniteurs de ressources que vous avez précédemment définis afin de pouvoir réutiliser leurs définitions ultérieurement, par exemple pour recréer les moniteurs de ressources dans une autre infrastructure ou si un moniteur de ressources doit être recréé en raison de problèmes liés au gestionnaire de files d'attente.

Vous pouvez sauvegarder une définition de gestionnaire de ressources unique à l'aide de la commande **fteCreateMonitor** ou de la commande **fteListMonitors** avec le paramètre **-ox**. Dans les deux cas, la définition du gestionnaire de ressources est sauvegardée en l'exportant dans un fichier XML. Vous pouvez ensuite utiliser le paramètre **-ix** de la commande **fteCreateMonitor** pour créer un gestionnaire de ressources en important la définition à partir du fichier XML.

Avec le paramètre **-ox**, vous ne pouvez sauvegarder qu'une seule définition de moniteur de ressources à la fois.

V 9.0.5 A partir de la IBM MQ 9.0.5, le paramètre **-od** est ajouté à la commande **fteListMonitors**. En le spécifiant, vous pouvez sauvegarder plusieurs moniteurs de ressources simultanément en exportant leurs définitions en bloc dans un répertoire spécifié. Chaque définition de moniteur de ressources est sauvegardée dans un fichier XML distinct avec un nom au format *agent name.monitor name.xml*.

V 9.0.5 Le paramètre **-od** est particulièrement utile si vous disposez d'un grand nombre de moniteurs de ressources que vous souhaitez sauvegarder car vous devez exécuter la commande **fteListMonitors -od** une seule fois, au lieu d'exécuter la commande **fteListMonitors -ox** séparément pour chaque définition de ressource, ou utiliser un script distinct pour exécuter la commande **fteListMonitors -ox** pour chaque moniteur de ressources.

Procédure

- Pour sauvegarder la définition d'un moniteur de ressources en l'exportant dans un fichier XML, utilisez l'une des commandes suivantes:
 - La commande **fteCreateMonitor** avec le paramètre **-ox**.
 - La commande **fteListMonitors** avec le paramètre **-ox**.

Lorsque vous utilisez le paramètre **-ox**, vous devez également spécifier les paramètres **-ma** et **-mn**, comme illustré dans l'exemple suivant:

```
fteListMonitors -ma AGENT1 -mn MONITOR1 -ox filename1.xml
```

- **V 9.0.5** Pour sauvegarder plusieurs définitions de moniteur de ressources en les exportant vers des fichiers XML dans un répertoire spécifié, utilisez la commande **fteListMonitors** avec le paramètre **-od**, comme illustré dans l'exemple suivant:

```
fteListMonitors -od /usr/mft/resmonbackup
```

Vous devez spécifier un répertoire cible valide lorsque vous sauvegardez des moniteurs de ressources en bloc. Si vous ne spécifiez pas de chemin cible, un message d'erreur est généré, comme illustré dans l'exemple suivant:

```
BFGCL0762E: Répertoire de sortie non spécifié. Réexécutez la commande en spécifiant un chemin valide.
```

Le paramètre **-od** ne doit pas être combiné avec le paramètre **-ox**, sinon le message d'erreur suivant s'affiche:

```
BFGCL0761E: Il n'est pas valide de spécifier à la fois les paramètres '-od'et'-ox'.
```

Vous pouvez définir un ensemble particulier de moniteurs de ressources à inclure dans la sauvegarde. Par exemple, en utilisant le paramètre **-ma** pour spécifier le nom d'un agent, vous pouvez sauvegarder tous les moniteurs de ressources de cet agent, comme illustré dans l'exemple suivant:

```
fteListMonitors -ma AGENT1 -od /usr/mft/resmonbackup
```

Vous pouvez également utiliser la mise en correspondance générique en incluant un astérisque (*) lors de la définition d'un modèle à utiliser pour la mise en correspondance des noms d'agent ou des noms de moniteur, ou les deux. L'exemple suivant sauvegarde tous les moniteurs de ressources dont les noms correspondent à un modèle spécifié et qui se trouvent dans un agent dont le nom correspond à un modèle spécifié:

```
fteListMonitors -ma AGENT* -mn MON* -od /usr/mft/resmonbackup
```

Lorsque la commande est en cours d'exécution, elle affiche les messages de rapport de progression suivants:

```
Un total de nombre de définitions de moniteur de ressources correspondantes a été trouvé.  
index de nombre définitions de moniteur de ressources sauvegardées dans le système de fichiers.
```

Si vous utilisez l'option prolixe, le total en cours d'exécution est toujours affiché, mais au lieu d'afficher *index* de *nombre* définitions de moniteur de ressources sauvegardées dans le système de fichiers la commande affiche le nom de la définition de moniteur en cours de sauvegarde, par exemple:

```
BFGCL0762I: Définition du moniteur'FILEMON'de l'agent'XFERAGENT'sauvegardé en tant que  
FILEMON.XFERAGENT.XML vers le système de fichiers.
```

V 9.0.5

Pour sauvegarder un moniteur de ressources pour un agent particulier en l'exportant dans un fichier XML dans un répertoire spécifié, utilisez la commande **fteListMonitors** avec le paramètre **-od** :

```
fteListMonitors -ma AGENT1 -mn MONITOR1 -od /usr/mft/resmonbackup
```

L'utilisation du paramètre **-od** pour sauvegarder un moniteur de ressources unique est similaire à l'utilisation du paramètre **-ox**, sauf que le nom du fichier de sortie est au format *agent.name.monitor.name.xml*.

- Pour restaurer des définitions de moniteur de ressources à partir d'une sauvegarde, utilisez la commande **fteCreateMonitor** avec le paramètre **-ix**, comme illustré dans l'exemple suivant:

```
fteCreateMonitor -ix file name
```

Pour plus d'exemples d'utilisation du paramètre **-od**, voir [«fteListMoniteurs: répertoire les moniteurs de ressources MFT»](#), à la page 487.

Référence associée

[«fteCreateMonitor: créer un moniteur de ressources MFT»](#), à la page 424

La commande **fteCreateMonitor** crée et démarre un nouveau moniteur de ressources à partir de la ligne de commande. Vous pouvez surveiller une ressource (par exemple, le contenu d'un répertoire) à l'aide de Managed File Transfer de sorte que lorsqu'une condition de déclenchement est remplie, une tâche spécifiée, telle qu'un transfert de fichier, est démarrée.

[«fteListMoniteurs: répertoire les moniteurs de ressources MFT»](#), à la page 487

La commande **fteListMonitors** permet de répertorier tous les moniteurs de ressources existants dans un réseau Managed File Transfer à l'aide de la ligne de commande.

Utilisation des modèles de transfert de fichiers

Vous pouvez utiliser des modèles de transfert de fichiers pour stocker des paramètres de transfert de fichiers communs pour des transferts répétés ou complexes. Créez un modèle de transfert à partir de la

ligne de commande à l'aide de la commande **fteCreateTemplate** ou utilisez la IBM MQ Explorer pour créer un modèle de transfert à l'aide de l'assistant **Créer un modèle pour Managed File Transfer** , ou sauvegardez un modèle lors de la création d'un transfert de fichier en cochant la case **Sauvegarder les paramètres de transfert en tant que modèle** . La fenêtre **Modèles de transfert** affiche tous les modèles de transfert que vous avez créés dans votre réseau Managed File Transfer .

Pourquoi et quand exécuter cette tâche

Pour créer un modèle de transfert à partir de la ligne de commande, utilisez la commande [fteCreateTemplate](#) . Ensuite, lorsque vous souhaitez soumettre un modèle de transfert que vous avez créé sur la ligne de commande, cliquez sur **Soumettre** dans IBM MQ Explorer.

Pour afficher les modèles de transfert dans le IBM MQ Explorer, procédez comme suit:

Procédure

1. Développez **Managed File Transfer** dans la vue Navigator . **Managed File Transfer Central** s'affiche dans la vue Contenu.
2. Tous les gestionnaires de files d'attente de coordination sont répertoriés dans la vue Navigator . Développez le nom du gestionnaire de file d'attente de coordination que vous avez utilisé pour le transfert planifié. Si vous souhaitez modifier le gestionnaire de file d'attente de coordination auquel vous êtes connecté, cliquez avec le bouton droit de la souris sur le nom du gestionnaire de file d'attente de coordination à utiliser dans la vue Navigator , puis cliquez sur **Connexion**.
3. Cliquez sur **Modèles de transfert**. La fenêtre **Modèles de transfert** s'affiche dans la vue Contenu.
4. La fenêtre **Modèles de transfert** répertorie les détails suivants sur vos transferts de fichiers:
 - a) **Nom** Nom de votre modèle de transfert de fichiers.
 - b) **Source** Nom de l'agent utilisé pour transférer le fichier à partir du système source.
 - c) **Fichier source** Nom du fichier à transférer sur son système hôte.
Développez les informations du modèle de transfert pour afficher cette zone.
 - d) **Destination** Nom de l'agent utilisé pour recevoir le fichier sur le système de destination.
 - e) **Fichier de destination** Nom du fichier après son transfert vers le système de destination.
Développez les informations du modèle de transfert pour afficher cette zone.
 - f) **Début planifié (fuseau horaire sélectionné)** Date et heure auxquelles le transfert de fichier est planifié pour démarrer dans le fuseau horaire utilisé par l'administrateur. Pour modifier le fuseau horaire affiché, cliquez sur **Fenêtre > Préférences > IBM MQ Explorer > Managed File Transfer** et sélectionnez un autre fuseau horaire dans la liste **Fuseau horaire:** . Cliquez sur **OK**.
 - g) **Événements déclencheurs** Type d'événement qui déclenche le démarrage du transfert de fichiers. Le type peut être l'une des valeurs suivantes: existe, n'existe pas ou dépasse.

Résultats

Pour actualiser les éléments affichés dans la fenêtre **Modèles de transfert** , cliquez sur le bouton Actualiser  dans la barre d'outils de la vue Contenu.

Pour soumettre un modèle de transfert et démarrer le transfert défini dans le modèle, cliquez avec le bouton droit de la souris sur le nom du modèle et cliquez sur **Soumettre**.

Pour modifier un modèle de transfert, cliquez avec le bouton droit de la souris sur le nom du modèle et cliquez sur **Editer**. Tous les fichiers inclus dans le modèle d'origine sont répertoriés comme faisant partie d'un groupe de transfert, même s'ils n'ont pas été inclus comme faisant partie d'un groupe dans le modèle d'origine. Si vous souhaitez supprimer un fichier du modèle, vous devez sélectionner la spécification de fichier dans le groupe et cliquer sur **Supprimer la sélection**. Si vous souhaitez ajouter de nouvelles spécifications de fichier au modèle, utilisez les zones du panneau Modèle et cliquez sur le bouton **Ajouter au groupe** . Une fois que vous avez effectué vos modifications, vous êtes invité à attribuer un nouveau nom au modèle édité.

Pour créer un transfert de fichier à partir d'un modèle de transfert, cliquez avec le bouton droit de la souris sur le nom du modèle et cliquez sur **Editer en tant que nouveau transfert**.

Pour créer une copie en double d'un modèle de transfert, cliquez avec le bouton droit de la souris sur le nom du modèle et cliquez sur **Dupliquer**. Le modèle de transfert en double est automatiquement sauvegardé avec le même nom que le modèle d'origine, auquel est ajouté "(copy)".

Pour supprimer un modèle de transfert, cliquez avec le bouton droit de la souris sur le nom du modèle, puis cliquez sur **Supprimer**.

Tâches associées

«Création d'un modèle de transfert de fichiers à l'aide de IBM MQ Explorer», à la page 202

Vous pouvez créer un modèle de transfert de fichier à partir de IBM MQ Explorer ou de la ligne de commande. Vous pouvez ensuite utiliser ce modèle pour créer de nouveaux transferts de fichiers à l'aide des détails du modèle ou soumettre le modèle pour démarrer le transfert de fichiers.

Référence associée

«fteCreateTemplate: créer un nouveau modèle de transfert de fichiers», à la page 431

La commande **fteCreateTemplate** crée un modèle de transfert de fichier que vous pouvez conserver pour une utilisation ultérieure. Le seul paramètre obligatoire est le paramètre **-tn** (*template_name*). Tous les autres paramètres sont facultatifs, bien que si vous spécifiez une spécification de fichier source, vous devez également fournir un fichier de destination. De même, si vous spécifiez un fichier de destination, vous devez également spécifier une spécification de fichier source.

«fteListTemplates», à la page 492

La commande **fteListTemplates** permet de répertorier les modèles de transfert Managed File Transfer disponibles sur un gestionnaire de files d'attente de coordination.

«fteDeleteTemplates», à la page 481

La commande **fteDeleteTemplates** permet de supprimer un modèle Managed File Transfer existant d'un gestionnaire de files d'attente de coordination.

Création d'un modèle de transfert de fichiers à l'aide de IBM MQ Explorer

Vous pouvez créer un modèle de transfert de fichier à partir de IBM MQ Explorer ou de la ligne de commande. Vous pouvez ensuite utiliser ce modèle pour créer de nouveaux transferts de fichiers à l'aide des détails du modèle ou soumettre le modèle pour démarrer le transfert de fichiers.

Pourquoi et quand exécuter cette tâche

Pour créer un modèle de transfert de fichier à partir de la ligne de commande, utilisez la commande [fteCreateTemplate](#).

Pour créer un modèle de transfert de fichiers à l'aide de l'assistant **Créer un modèle pour Managed File Transfer** dans IBM MQ Explorer, procédez comme suit:

Procédure

1. Dans la vue Navigator, cliquez sur **Transfert de fichier géré. Managed File Transfer Central** s'affiche dans la vue Contenu.
2. Tous vos gestionnaires de files d'attente de coordination sont affichés dans la vue Navigator. Développez le nom du gestionnaire de file d'attente de coordination que vous avez utilisé pour le transfert planifié. Si vous souhaitez modifier le gestionnaire de file d'attente de coordination auquel vous êtes connecté, cliquez avec le bouton droit de la souris sur le nom du gestionnaire de file d'attente de coordination à utiliser dans la vue Navigator, puis cliquez sur **Connexion**.
3. Démarrez l'assistant **Créer un modèle pour Managed File Transfer** en cliquant avec le bouton droit de la souris sur **Modèles de transfert**, puis en cliquant sur **Nouveau modèle**.
4. Suivez les instructions des panneaux de l'assistant. Une aide contextuelle est fournie pour chaque panneau. Pour accéder à l'aide contextuelle sur Windows, appuyez sur F1. Sous Linux, appuyez sur Ctrl+F1 ou sur Shift+F1.

Si vous avez créé un modèle contenant tous les détails de transfert requis, veillez à cocher la case **Sauvegarder les paramètres de transfert en tant que modèle** sur la page **Récapitulatif du transfert** si cette case n'est pas déjà cochée. Entrez également un nom pour le modèle dans la zone Nom. Si vous créez un modèle qui ne contient pas encore tous les détails de transfert requis, la case **Sauvegarder les paramètres de transfert en tant que modèle** est automatiquement cochée pour vous.

Tâches associées

«Utilisation des modèles de transfert de fichiers», à la page 200

Vous pouvez utiliser des modèles de transfert de fichiers pour stocker des paramètres de transfert de fichiers communs pour des transferts répétés ou complexes. Créez un modèle de transfert à partir de la ligne de commande à l'aide de la commande **fteCreateTemplate** ou utilisez la IBM MQ Explorer pour créer un modèle de transfert à l'aide de l'assistant **Créer un modèle pour Managed File Transfer**, ou sauvegardez un modèle lors de la création d'un transfert de fichier en cochant la case **Sauvegarder les paramètres de transfert en tant que modèle**. La fenêtre **Modèles de transfert** affiche tous les modèles de transfert que vous avez créés dans votre réseau Managed File Transfer.

Référence associée

«fteCreateTemplate: créer un nouveau modèle de transfert de fichiers», à la page 431

La commande **fteCreateTemplate** crée un modèle de transfert de fichier que vous pouvez conserver pour une utilisation ultérieure. Le seul paramètre obligatoire est le paramètre **-tn** (*template_name*). Tous les autres paramètres sont facultatifs, bien que si vous spécifiez une spécification de fichier source, vous devez également fournir un fichier de destination. De même, si vous spécifiez un fichier de destination, vous devez également spécifier une spécification de fichier source.

«fteListTemplates», à la page 492

La commande **fteListTemplates** permet de répertorier les modèles de transfert Managed File Transfer disponibles sur un gestionnaire de files d'attente de coordination.

«fteDeleteTemplates», à la page 481

La commande **fteDeleteTemplates** permet de supprimer un modèle Managed File Transfer existant d'un gestionnaire de files d'attente de coordination.

Transfert de données de fichiers vers des messages

Vous pouvez utiliser la fonction de fichier à message de Managed File Transfer pour transférer des données d'un fichier vers un ou plusieurs messages d'une file d'attente IBM MQ.

Pour effectuer des transferts de fichier à message et de message à fichier, l'agent source et l'agent de destination du transfert doivent être au niveau IBM WebSphere MQ 7.5 ou ultérieur, ou au niveau IBM WebSphere MQ File Transfer Edition 7.0.3 ou ultérieur. Pour plus d'informations sur les transferts de messages vers des fichiers, voir «[Transfert de données de messages vers des fichiers](#)», à la page 211.

L'agent de destination pour un transfert de fichier à message ne peut pas être un agent de pont de protocole ou un agent de pont Connect:Direct.

Vous pouvez transférer des données de fichier vers des données de message IBM MQ. Les messages IBM MQ peuvent être lus et utilisés par les applications. Les types de transfert de fichier à message suivants sont pris en charge:

- D'un fichier unique à un message unique. Aucun ID groupe IBM MQ n'est défini pour le message.
- D'un fichier unique à plusieurs messages, en fractionnant le fichier en messages d'une longueur donnée. Les messages ont tous le même ID groupe IBM MQ.
- D'un fichier unique à plusieurs messages, en fractionnant un fichier texte au niveau d'un délimiteur d'expression régulière Java. Les messages ont tous le même ID groupe IBM MQ.
- D'un fichier unique à plusieurs messages, en fractionnant un fichier binaire à l'aide d'un délimiteur hexadécimal. Les messages ont tous le même ID groupe IBM MQ.

Si vous souhaitez fractionner un fichier binaire en utilisant une séquence d'octets comme délimiteur, utilisez le paramètre **-sqdb** de la commande **fteCreateTransfer**. Pour plus d'informations, voir [Paramètre -sqdb](#).

Par défaut, les messages créés par un transfert de fichier à message sont persistants. Les messages peuvent être définis pour être non persistants ou pour avoir la valeur de persistance définie par la file d'attente de destination.

Si vous indiquez qu'un fichier est divisé en plusieurs messages, tous les messages créés à partir du fichier ont le même ID de groupe IBM MQ . Si vous ne spécifiez pas qu'un fichier est divisé en plusieurs messages, un seul message est créé à partir du fichier et l'ID de groupe IBM MQ n'est pas défini pour ce message.

Si vous transférez des fichiers vers des messages volumineux ou de nombreux messages de petite taille, vous devrez peut-être modifier certaines propriétés IBM MQ ou Managed File Transfer . Pour plus d'informations, voir [«Conseils pour la définition des attributs MQ et des propriétés MFT associées à la taille des messages»](#), à la page 329.

Remarque : Si la file d'attente de destination est une file d'attente en cluster ou un alias d'une file d'attente en cluster, vous obtiendrez un message d'erreur lors du transfert d'un fichier dans une file d'attente si la propriété d'agent `enableClusterQueueInputOutput` n'a pas été définie sur `true`. Pour plus d'informations, voir [«Que faire si la file d'attente de destination est une file d'attente en cluster ou un alias d'une file d'attente en cluster»](#), à la page 325

Configuration d'un agent pour effectuer des transferts de fichier à message

Par défaut, les agents ne peuvent pas effectuer de transferts de fichier à message ou de message à fichier. Pour activer cette fonction, vous devez définir la propriété d'agent `enableQueueInputOutput` sur `true`. Pour activer l'écriture dans les files d'attente en cluster IBM MQ , vous devez également définir la propriété d'agent `enableClusterQueueInputOutput` sur `true`.

Pourquoi et quand exécuter cette tâche

Si vous tentez d'effectuer un transfert de fichier à message vers un agent de destination dont la propriété `enableQueueInputOutput` n'est pas définie sur `true`, le transfert échoue. Le message de journal de transfert publié dans le gestionnaire de file d'attente de coordination contient le message suivant:

```
BFGI00197E: An attempt to write to a queue was rejected by the destination agent. The agent must have enableQueueInputOutput=true set in the agent.properties file to support transferring to a queue.
```

Pour permettre à l'agent d'écrire et de lire dans les files d'attente, procédez comme suit:

Procédure

1. Arrêtez l'agent de destination à l'aide de la commande **fteStopAgent** .
2. Editez le fichier `agent.properties` pour inclure la ligne `enableQueueInputOutput=true`.
Le fichier `agent.properties` se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/destination_agent_name`.
3. Facultatif : Editez le fichier `agent.properties` pour inclure la ligne `enableClusterQueueInputOutput=true`. Le fichier `agent.properties` se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/destination_agent_name`.
4. Démarrez l'agent cible à l'aide de la commande **fteStartAgent** .

Exemple: Transfert d'un fichier unique vers un message unique

Vous pouvez spécifier une file d'attente comme destination d'un transfert de fichier à l'aide du paramètre **-dq** avec la commande **fteCreateTransfer** . Le fichier source doit être inférieur à la longueur de message maximale définie dans la file d'attente de destination. La file d'attente de destination ne doit pas nécessairement se trouver sur le même gestionnaire de files d'attente que le gestionnaire de files d'attente auquel l'agent de destination se connecte, mais ces deux gestionnaires de files d'attente doivent pouvoir communiquer.

Pourquoi et quand exécuter cette tâche

Le fichier source est appelé `/tmp/single_record.txt` et se trouve sur le même système que l'agent source, AGENT_NEPTUNE. L'agent source, AGENT_NEPTUNE, utilise le gestionnaire de files d'attente QM_NEPTUNE. L'agent de destination est AGENT_VENUS et cet agent se connecte au gestionnaire de files d'attente QM_VENUS. La file d'attente de destination, RECEIVING_QUEUE, se trouve sur le gestionnaire de files d'attente QM_MERCURY. QM_MERCURY se trouve dans le même réseau IBM MQ que le gestionnaire de files d'attente QM_VENUS, auquel il peut accéder.

Procédure

Entrez la commande suivante :

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_VENUS -dm QM_VENUS  
-dq RECEIVING_QUEUE@QM_MERCURY /tmp/single_record.txt
```

Si la file d'attente de destination se trouve sur un gestionnaire de files d'attente différent de celui utilisé par l'agent de destination, vous devez spécifier la valeur du paramètre **-dq** au format suivant: *nom_file d'attente@nom_gestionnaire_files d'attente*. Si vous ne spécifiez pas *@nom_gestionnaire_files_attente* dans la valeur, l'agent de destination suppose que la file d'attente de destination se trouve sur le gestionnaire de files d'attente de l'agent de destination. La seule exception est lorsque la propriété de l'agent de sortie `enableClusterQueueInputa` a été définie sur `true`. Dans ce cas, l'agent de destination utilise des procédures de résolution IBM MQ standard pour déterminer où se trouve la file d'attente.

L'agent source, AGENT_NEPTUNE, lit les données du fichier `/tmp/single_record.txt` et les transfère à l'agent de destination, AGENT_VENUS. L'agent de destination, AGENT_VENUS, envoie les données à un message persistant dans la file d'attente `RECEIVING_QUEUE@QM_MERCURY`. Aucun ID groupe IBM MQ n'est défini pour le message.

Exemple: Fractionnement d'un fichier unique en plusieurs messages par longueur

Vous pouvez fractionner un fichier en plusieurs messages IBM MQ à l'aide du paramètre **-qs** de la commande **fteCreateTransfer**. Le fichier est divisé en sections de longueur fixe, dont chacune est écrite dans un message individuel.

Pourquoi et quand exécuter cette tâche

Le fichier source est appelé `/tmp/source.file` et sa taille est de 36 Ko. Le fichier source se trouve sur le même système que l'agent source AGENT_NEPTUNE. L'agent source, AGENT_NEPTUNE, se connecte au gestionnaire de files d'attente QM_NEPTUNE. L'agent de destination est AGENT_MERCURY, qui se connecte au gestionnaire de files d'attente QM_MERCURY. La file d'attente de destination, RECEIVING_QUEUE, se trouve également sur le gestionnaire de files d'attente QM_MERCURY. Le transfert fractionne le fichier source en sections d'une taille de 1 Ko et écrit chacune de ces sections dans un message sur RECEIVING_QUEUE.

Procédure

Entrez la commande suivante :

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_MERCURY -dm QM_MERCURY  
-dq RECEIVING_QUEUE -qs 1K /tmp/source.file
```

L'agent source, AGENT_NEPTUNE, lit les données du fichier `/tmp/source.file` et les transfère à l'agent cible, AGENT_MERCURY. L'agent de destination, AGENT_MERCURY, écrit les données dans trente-six messages persistants de 1 ko dans la file d'attente `RECEIVING_QUEUE@QM_MERCURY`. Ces messages ont tous le même ID de groupe IBM MQ et l'indicateur IBM MQ `LAST_MSG_IN_GROUP` est défini pour le dernier message du groupe.

Exemple: division d'un fichier texte en plusieurs messages à l'aide d'un délimiteur d'expression régulière

Transférez un fichier texte unique dans plusieurs messages en fractionnant le fichier à chaque correspondance d'une expression régulière Java donnée. Pour ce faire, utilisez le paramètre **-dqdt** de la commande **fteCreateTransfer**.

Pourquoi et quand exécuter cette tâche

Le fichier est divisé en sections de longueur variable, dont chacune est écrite dans un message individuel. Le fichier texte est fractionné à chaque point où le texte du fichier correspond à une expression régulière donnée. Le fichier source est appelé `/tmp/names.text` et a le contenu suivant:

```
Jenny Jones,John Smith,Jane Brown
```

L'expression régulière qui indique où scinder le fichier est la virgule (`,`).

Le fichier source se trouve sur le même système que l'agent source `AGENT_NEPTUNE`, qui se connecte au gestionnaire de files d'attente `QM_NEPTUNE`. La file d'attente de destination, `RECEIVING_QUEUE`, se trouve sur le gestionnaire de files d'attente `QM_MERCURY`. `QM_MERCURY` est également le gestionnaire de files d'attente utilisé par l'agent de destination `AGENT_MERCURY`. Le transfert fractionne le fichier source en sections et écrit chacune de ces sections dans un message sur `RECEIVING_QUEUE`.

Procédure

Entrez la commande suivante :

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_MERCURY -dm QM_MERCURY  
-dq RECEIVING_QUEUE -t text -dqdp postfix -dqdt "," /tmp/names.text
```

L'agent source, `AGENT_NEPTUNE`, lit les données du fichier `/tmp/names.text` et les transfère à l'agent cible, `AGENT_MERCURY`. L'agent de destination, `AGENT_MERCURY`, écrit les données dans trois messages persistants de la file d'attente `RECEIVING_QUEUE`. Ces messages ont tous le même ID de groupe IBM MQ et l'indicateur IBM MQ `LAST_MSG_IN_GROUP` est défini pour le dernier message du groupe.

Les données des messages sont les suivantes.

- Premier message:

```
Jenny Jones
```

- Deuxième message:

```
John Smith
```

- Troisième message:

```
Jane Brown
```

Exemple: Fractionnement d'un fichier texte avec un délimiteur d'expression régulière et inclusion du délimiteur dans les messages

Transférez un fichier texte unique dans plusieurs messages en fractionnant le fichier à chaque correspondance d'une expression régulière Java donnée et incluez la correspondance d'expression régulière dans les messages résultants. Pour ce faire, utilisez les paramètres **-dqdt** et **-qi** de la commande **fteCreateTransfer**.

Pourquoi et quand exécuter cette tâche

Transférez un fichier texte unique vers plusieurs messages d'une file d'attente. Le fichier est divisé en sections de longueur variable, dont chacune est écrite dans un message individuel. Le fichier texte est fractionné à chaque point où le texte du fichier correspond à une expression régulière donnée. Le fichier source est appelé `/tmp/customers.text` et a le contenu suivant:

```
Customer name: John Smith
Customer contact details: john@example.net
Customer number: 314

Customer name: Jane Brown
Customer contact details: jane@example.com
Customer number: 42

Customer name: James Jones
Customer contact details: jjones@example.net
Customer number: 26
```

L'expression régulière qui indique où scinder le fichier est `Customer\snumber:\s\d+`, qui correspond au texte "Customer number: " suivi de n'importe quel nombre de chiffres. Les expressions régulières spécifiées sur la ligne de commande doivent être placées entre guillemets pour empêcher le shell de commandes d'évaluer l'expression régulière. L'expression régulière est évaluée en tant qu'expression régulière Java. Pour plus d'informations, voir la section [«Expressions régulières utilisées par MFT»](#), à la page 679.

Par défaut, le nombre de caractères pouvant correspondre à une expression régulière est défini sur cinq. L'expression régulière utilisée dans cet exemple correspond à des chaînes de plus de cinq caractères. Pour activer les correspondances de plus de cinq caractères, éditez le fichier de propriétés de l'agent pour inclure la propriété **maxDelimiterMatchLength**.

Par défaut, le texte qui correspond à l'expression régulière n'est pas inclus dans les messages. Pour inclure le texte qui correspond à l'expression régulière dans les messages, comme dans cet exemple, utilisez le paramètre **-qi**. Le fichier source se trouve sur le même système que l'agent source AGENT_NEPTUNE, qui se connecte au gestionnaire de files d'attente QM_NEPTUNE. La file d'attente de destination, RECEIVING_QUEUE, se trouve sur le gestionnaire de files d'attente QM_MERCURY. QM_MERCURY est également le gestionnaire de files d'attente utilisé par l'agent de destination AGENT_MERCURY. Le transfert fractionne le fichier source en sections et écrit chacune de ces sections dans un message sur RECEIVING_QUEUE.

Procédure

1. Arrêtez l'agent de destination à l'aide de la commande suivante:

```
fteStopAgent AGENT_MERCURY
```

2. Ajoutez la ligne suivante au fichier de propriétés de l'agent pour AGENT_MERCURY:

```
maxDelimiterMatchLength=25
```

Remarque : L'augmentation de la valeur de **maxDelimiterMatchLength** peut réduire les performances.

3. Démarrez l'agent cible à l'aide de la commande suivante:

```
fteStartAgent AGENT_MERCURY
```

4. Entrez la commande suivante :

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_MERCURY -dm QM_MERCURY
-dq RECEIVING_QUEUE
text -dqdt "Customer\snumber:\s\d+" -qi -dqdp postfix /tmp/customers.text
```

L'agent source, AGENT_NEPTUNE, lit les données du fichier /tmp/customers.text et les transfère à l'agent cible, AGENT_MERCURY. L'agent de destination, AGENT_MERCURY, écrit les données dans trois messages persistants de la file d'attente RECEIVING_QUEUE. Ces messages ont tous le même ID de groupe IBM MQ et l'indicateur IBM MQ LAST_MSG_IN_GROUP est défini pour le dernier message du groupe.

Les données des messages sont les suivantes.

- Premier message:

```
Customer name: John Smith
Customer contact details: john@example.net
Customer number: 314
```

- Deuxième message:

```
Customer name: Jane Brown
Customer contact details: jane@example.com
Customer number: 42
```

- Troisième message:

```
Customer name: James Jones
Customer contact details: jjones@example.net
Customer number: 26
```

Exemple: Définition des propriétés de message IBM MQ sur un transfert de fichier à message

Vous pouvez utiliser le paramètre **-qmp** dans la commande **fteCreateTransfer** pour indiquer si les propriétés de message IBM MQ sont définies sur le premier message écrit dans la file d'attente de destination par le transfert. Les propriétés de message IBM MQ permettent à une application de sélectionner des messages à traiter ou d'extraire des informations sur un message sans accéder aux en-têtes IBM MQ Message Descriptor (MQMD) ou MQRFH2 .

Pourquoi et quand exécuter cette tâche

Incluez le paramètre **-qmp true** dans la commande **fteCreateTransfer** . Dans cet exemple, l'ID utilisateur MQMD de l'utilisateur qui soumet la commande est **larmer**.

Procédure

Entrez la commande suivante :

```
fteCreateTransfer -sa AGENT_JUPITER -da AGENT_SATURN -dq MY_QUEUE@MyQM -qmp true
-t text /tmp/source_file.txt
```

Les propriétés de message IBM MQ du premier message écrit par l'agent de destination, AGENT_SATURN, dans la file d'attente, MY_QUEUE, sur le gestionnaire de files d'attente, MyQM, sont définies sur les valeurs suivantes:

```
usr.WMQFTETransferId=414cbaedefa234889d999a8ed09782395ea213ebbc9377cd
usr.WMQFTETransferMode=text
usr.WMQFTESourceAgent=AGENT_JUPITER
usr.WMQFTEDestinationAgent=AGENT_SATURN
usr.WMQFTEFileName=source_file.txt
usr.WMQFTEFileSize=1024
usr.WMQFTEFileLastModified=1273740879040
usr.WMQFTEFileIndex=0
usr.WMQFTEMQMDUser=larmer
```

Exemple: Définition de propriétés définies par l'utilisateur sur un transfert de fichier à message

Les métadonnées définies par l'utilisateur sont définies en tant que propriété de message IBM MQ sur le premier message écrit dans la file d'attente de destination par le transfert. Les propriétés de message IBM MQ permettent à une application de sélectionner des messages à traiter ou d'extraire des informations sur un message sans accéder aux en-têtes IBM MQ Message Descriptor (MQMD) ou MQRFH2.

Pourquoi et quand exécuter cette tâche

Incluez les paramètres `-qmp true` et `-md account=123456` dans la commande `fteCreateTransfer` pour définir la propriété `usr.account` sur 123456 dans l'en-tête RFH2.

Procédure

Entrez la commande suivante :

```
fteCreateTransfer -sa AGENT_JUPITER -da AGENT_SATURN -dq MY_QUEUE@MyQM
-qmp true -md account=123456 /tmp/source_file.txt
```

Outre l'ensemble standard de propriétés de message IBM MQ, la propriété définie par l'utilisateur est définie dans l'en-tête du premier message écrit par l'agent de destination, AGENT_SATURN, dans la file d'attente, MY_QUEUE, sur le gestionnaire de files d'attente, MyQM. L'en-tête est défini sur la valeur suivante:

```
usr.account=123456
```

Le préfixe `usr` est ajouté au début du nom des métadonnées définies par l'utilisateur.

Exemple: ajout d'une propriété de message définie par l'utilisateur pour un transfert de fichier à message

Si vous utilisez Managed File Transfer pour les transferts gérés de message à fichier, vous pouvez inclure une propriété de message définie par l'utilisateur pour le message résultant.

Pourquoi et quand exécuter cette tâche

Vous pouvez utiliser l'une des méthodes suivantes pour définir une propriété de message personnalisée:

- Indiquez le paramètre `-md` dans la demande de transfert. Pour plus d'informations, voir la section [«Exemple: Définition de propriétés définies par l'utilisateur sur un transfert de fichier à message», à la page 209.](#)
- Utilisez une tâche Ant ; vous pouvez utiliser `fte:filecopy` ou `fte:filemove`. L'exemple suivant est une tâche `fte:filecopy`:

```
<project xmlns:fte="antlib:com.ibm.wmqfte.ant.taskdefs" default="complete">
<!-- Initialise the properties used in this script.-->

<target name="init" description="initialise task properties">
  <property name="src.file" value="/home/user/file1.bin"/>
  <property name="dst.queue" value="TEST.QUEUE@qm2"/>
  <fte:uuid property="job.name" length="8"
prefix="copyjob#"/>
</target>
<target name="step1" depends="init" description="transfer file">

<fte:filecopy cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
src="agent1@qm1" dst="agent2@qm2"
rcproperty="copy.result">

<fte:metadata>
```

```

<fte:entry name="fileName" value="{FileName}"/>
</fte:metadata>

<fte:filespec srcfilespec="{src.file}" dstqueue="{dst.queue}"
dstmsgprops="true"/>

</fte:filecopy>

</target>
</project>

```

- Utilisez un moniteur de ressources et une substitution de variable. L'exemple suivant illustre un code XML de tâche de transfert:

```

<?xml version="1.0" encoding="UTF-8"?>
<monitor:monitor
xmlns:monitor="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance" version="5.00"
xsi:schemaLocation="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition ./Monitor.xsd">
  <name>METADATA</name>
  <pollInterval units="minutes">5</pollInterval>
  <batch maxSize="5"/>
  <agent>AGENT1</agent>
  <resources>
    <directory recursionLevel="0">e:\temp</directory>
  </resources>
  <triggerMatch>
    <conditions>
      <allOf>
        <condition>
          <fileMatch>
            <pattern>*.txt</pattern>
          </fileMatch>
        </condition>
      </allOf>
    </conditions>
  </triggerMatch>
  <tasks>
    <task>
      <name/>
      <transfer>
        <request version="5.00"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
          <managedTransfer>
            <originator>
              <hostName>mqjason.raleigh.ibm.com.</hostName>
              <userID>administrator</userID>
            </originator>
            <sourceAgent QMgr="AGENTQM" agent="AGENT1"/>
            <destinationAgent QMgr="AGENTQM" agent="AGENT2"/>
            <transferSet priority="0">
              <metaDataSet>
                <metaData key="FileName">{FileName}</metaData>
              </metaDataSet>
              <item checksumMethod="MD5" mode="text">
                <source disposition="delete" recursive="false">
                  <file>{FilePath}</file>
                </source>
                <destination type="queue">
                  <queue persistent="true"
setMqProps="true">TEST.QUEUE@AGENTQM</queue>
                </destination>
              </item>
            </transferSet>
          </job>
        </managedTransfer>
      </request>
    </transfer>
  </task>
</tasks>
<originator>
  <hostName>mqjason.raleigh.ibm.com.</hostName>
  <userID>administrator</userID>

```

```
</originator>  
</monitor:monitor>
```

Tâches associées

«[Exemple: Définition des propriétés de message IBM MQ sur un transfert de fichier à message](#)», à la page 208

Vous pouvez utiliser le paramètre **-qmp** dans la commande **fteCreateTransfer** pour indiquer si les propriétés de message IBM MQ sont définies sur le premier message écrit dans la file d'attente de destination par le transfert. Les propriétés de message IBM MQ permettent à une application de sélectionner des messages à traiter ou d'extraire des informations sur un message sans accéder aux en-têtes IBM MQ Message Descriptor (MQMD) ou MQRFH2.

Référence associée

«[fte: tâche Ant filecopy](#)», à la page 878

La tâche **fte:filecopy** copie les fichiers entre les agents Managed File Transfer. Le fichier n'est pas supprimé de l'agent source.

«[fte: tâche Ant filemove](#)», à la page 881

La tâche **fte:filemove** déplace les fichiers entre les agents Managed File Transfer. Lorsqu'un fichier a été transféré avec succès de l'agent source vers l'agent cible, le fichier est supprimé de l'agent source.

Echec d'un transfert de fichier à message

Si un transfert de fichier à message échoue après que l'agent a commencé à écrire des données de fichier dans la file d'attente de destination, l'agent écrit un message dans la file d'attente pour indiquer à une application consommant les messages qu'un incident s'est produit.

Message écrit dans la file d'attente de destination en cas d'échec:

- Est vide
- A le même ID groupe IBM MQ que le message précédent écrit dans la file d'attente de destination par l'agent
- A l'indicateur IBM MQ LAST_MSG_IN_GROUP défini
- Contient des propriétés de message IBM MQ supplémentaires, si les propriétés de message sont activées. Pour plus d'informations, voir «[Propriétés d'échec](#)», à la page 699.

Exemple

Un transfert est demandé en exécutant la commande suivante:

```
fteCreateTransfer -sa AGENT_JUPITER -da AGENT_SATURN -dq RECEIVING_QUEUE  
-qmp true -qs 1K /tmp/source1.txt
```

Le fichier `source1.txt` est de 48 Ko. Le transfert fractionne ce fichier en messages de 1 Ko et écrit ces messages dans la file d'attente de destination `RECEIVING_QUEUE`.

Pendant que le transfert est en cours, une fois que l'agent a écrit 16 messages dans `RECEIVING_QUEUE`, un échec se produit au niveau de l'agent source.

L'agent écrit un message vide dans `RECEIVING_QUEUE`. En plus de l'ensemble standard de propriétés de message, le message vide possède les propriétés de message suivantes:

```
usr.WMQFTEResultCode = 40  
usr.WMQFTESupplement = BFGTR0036I: The transfer failed to complete successfully.
```

Transfert de données de messages vers des fichiers

La fonction de message à fichier de Managed File Transfer vous permet de transférer des données d'un ou de plusieurs messages d'une file d'attente IBM MQ vers un fichier, un fichier (sous z/OS), ou un espace fichier utilisateur. Si vous disposez d'une application qui crée ou traite des messages IBM MQ, vous

pouvez utiliser la fonction de message à fichier de Managed File Transfer pour transférer ces messages vers un fichier sur n'importe quel système de votre réseau Managed File Transfer .

Pour plus d'informations sur les transferts de fichier à message, voir [«Transfert de données de fichiers vers des messages»](#), à la page 203.



Avertissement : L'agent source d'un transfert de message à fichier ne peut pas être un agent de pont de protocole ou un agent de pont Connect:Direct .

Vous pouvez transférer des données de message IBM MQ dans un fichier. Les types de transfert de message vers fichier suivants sont pris en charge:

- D'un message unique vers un fichier unique
- De plusieurs messages vers un seul fichier
- De plusieurs messages avec le même ID groupe IBM MQ vers un seul fichier.
- De plusieurs messages à un seul fichier, y compris un délimiteur texte ou binaire entre les données de chaque message écrit dans le fichier.

Si vous transférez des fichiers à partir de messages volumineux ou de nombreux messages de petite taille, vous devrez peut-être modifier certaines propriétés IBM MQ ou Managed File Transfer . Pour plus d'informations, voir [«Conseils pour la définition des attributs MQ et des propriétés MFT associées à la taille des messages»](#), à la page 329.

V 9.0.1

Depuis IBM MQ 9.0.1, dans un transfert de message à fichier, l'agent source parcourt les messages de la file d'attente source, contrairement à l'opération GET destructive, dans les versions IBM MQ antérieures à IBM MQ 9.0.1. Les messages sont supprimés de la file d'attente source une fois que tous les messages (d'un groupe si le regroupement de messages est utilisé) ont été parcourus et que les données ont été écrites dans le fichier de destination. Ainsi, les messages peuvent rester dans la file d'attente source si un transfert échoue ou est annulé. En raison de cette modification, le droit de parcourir (BROWSE) doit également être fourni avec le droit d'obtention (GET) pour l'exécution de transferts de messages dans des fichiers.

V 9.0.0.2

Depuis la IBM MQ 9.0.0 Fix Pack 2, Managed File Transfer a été mis à jour pour restaurer la vérification de comparaison, précédemment supprimée par le correctif [APAR IT18213](#), de l'identificateur et la valeur de transfert de l'attribut groupId dans la charge XML des demandes de transfert. Si ces deux identificateurs sont équivalents, l'agent source utilise l'identificateur comme option de correspondance des identificateurs de message (et non comme option de correspondance des identificateurs de groupe) pour la première tentative MQGET effectuée sur la file d'attente d'entrée du transfert de message à fichier.

Configuration d'un agent pour effectuer des transferts de messages vers des fichiers

Par défaut, les agents ne peuvent pas effectuer de transferts de message à fichier ou de fichier à message. Pour activer cette fonction, vous devez définir la propriété d'agent `enableQueueInputOutput` sur `true`.

Pourquoi et quand exécuter cette tâche

Si vous tentez d'exécuter un message de transfert de fichier à partir d'un agent source dont la propriété `enableQueueInputOutput` n'est pas définie sur `true`, le transfert échoue. Le message de journal de transfert publié dans le gestionnaire de file d'attente de coordination contient le message suivant:

```
BFGI00197E: An attempt to read from a queue was rejected by the source agent.
The agent must have enableQueueInputOutput=true set in the agent.properties file
to support transferring from a queue.
```

Pour permettre à l'agent d'écrire et de lire dans les files d'attente, procédez comme suit:

Exemple: Insertion d'un délimiteur de texte avant les données de chaque message

Lorsque vous effectuez un transfert en mode texte d'une file d'attente source vers un fichier, vous pouvez indiquer qu'un délimiteur de texte est inséré avant les données des messages individuels à l'aide des paramètres **-sq**, **-sqdt** et **-sqdp** avec la commande **fteCreateTransfer**.

Pourquoi et quand exécuter cette tâche

Dans cet exemple, il y a quatre messages dans la file d'attente `START_QUEUE`. Cette file d'attente se trouve sur le gestionnaire de files d'attente de l'agent source, `QM_NEPTUNE`. Le délimiteur de texte à insérer avant les données de chaque message peut être exprimé sous la forme d'une chaîne littérale Java, par exemple: `\n\u002D\u002D\u002D\n`.

Procédure

Entrez la commande suivante :

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_VENUS -df /out/output.txt  
-t text -sqdt "\n\u002D\u002D\u002D\n" -sqdp prefix -sq START_QUEUE
```

Le délimiteur de texte est ajouté au début des données de chacun des quatre messages de `START_QUEUE` par l'agent source, `AGENT_NEPTUNE`. Ces données sont écrites dans le fichier de destination, `/out/output.txt`.

Exemple: Insertion d'un délimiteur binaire après les données de chaque message

Lors du transfert en mode binaire d'une file d'attente source vers un fichier, vous pouvez indiquer qu'un délimiteur binaire est inséré après les données des messages individuels à l'aide des paramètres **-sq**, **-sqdbet** et **-sqdp** avec la commande **fteCreateTransfer**.

Pourquoi et quand exécuter cette tâche

Dans cet exemple, il y a trois messages dans la file d'attente `START_QUEUE`. Cette file d'attente se trouve sur le gestionnaire de files d'attente de l'agent source, `QM_NEPTUNE`. Le délimiteur binaire à insérer après les données de chaque message doit être exprimé sous la forme d'une liste d'octets hexadécimaux séparés par des virgules, par exemple: `x34,xE7,xAE`.

Procédure

Entrez la commande suivante :

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_VENUS -df /out/binary.file  
-sqdp postfix -sqdb x34,xE7,xAE -sq START_QUEUE
```

Le délimiteur binaire est ajouté aux données de chacun des trois messages de `START_QUEUE` par l'agent source, `AGENT_NEPTUNE`. Ces données sont écrites dans le fichier de destination, `/out/binary.file`.

Surveillance d'une file d'attente et utilisation de la substitution de variable

Vous pouvez surveiller une file d'attente et transférer des messages de la file d'attente surveillée vers un fichier à l'aide de la commande **fteCreateMonitor**. La valeur de toute propriété de message IBM MQ dans le premier message à lire dans la file d'attente surveillée peut être remplacée dans la définition XML de tâche et utilisée pour définir le comportement de transfert.

Pourquoi et quand exécuter cette tâche

Dans cet exemple, l'agent source est appelé AGENT_VENUS, qui se connecte à QM_VENUS. La file d'attente surveillée par AGENT_VENUS est appelée START_QUEUE et se trouve sur QM_VENUS. L'agent interroge la file d'attente toutes les 30 minutes.

Lorsqu'un groupe complet de messages est écrit dans la file d'attente, la tâche de surveillance envoie le groupe de messages dans un fichier à l'un des agents de destination, qui se connectent tous au gestionnaire de files d'attente QM_MARS. Le nom du fichier vers lequel le groupe de messages est transféré est défini par la IBM MQ propriété de message `usr.fileName` sur le premier message du groupe. Le nom de l'agent auquel le groupe de messages est envoyé est défini par la IBM MQ propriété de message `usr.toAgent` sur le premier message du groupe. Si l'en-tête `usr.toAgent` n'est pas défini, la valeur par défaut à utiliser pour l'agent de destination est AGENT_MAGENTA.

Lorsque vous spécifiez `useGroups="true"`, si vous ne spécifiez pas également `groupId="{GROUPID}"`, le transfert prend simplement le premier message de la file d'attente. Par exemple, si vous utilisez la substitution de variable pour générer le `fileName`, il est possible que le contenu de `a.txt` ne soit pas correct. En effet, `fileName` est généré par le moniteur, mais le transfert reçoit en fait un message qui n'est pas celui qui doit générer le fichier appelé `fileName`.

Procédure

1. Créez le code XML de tâche qui définit la tâche que le moniteur exécute lorsqu'il est déclenché.

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_VENUS" QMgr="QM_VENUS"/>
    <destinationAgent agent="{toAgent}" QMgr="QM_MARS"/>
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <queue useGroups="true" groupId="{GROUPID}">START_QUEUE</queue>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/{fileName}.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Les variables qui sont remplacées par les valeurs des en-têtes de message IBM MQ sont mises en évidence en **gras**. Cette tâche XML est sauvegardée dans le fichier `/home/USER1/task.xml`

2. Créez un moniteur de ressources pour surveiller la file d'attente START_QUEUE.

Soumettez la commande suivante :

```
fteCreateMonitor -ma AGENT_VENUS -mm QM_VENUS -mq START_QUEUE
                 -mn myMonitor -mt /home/USER1/task.xml
                 -tr completeGroups -pi 30 -pu minutes -dv toAgent=AGENT_MAGENTA
```

3. Un utilisateur ou un programme écrit un groupe de messages dans la file d'attente START_QUEUE. Les propriétés de message IBM MQ suivantes sont définies pour le premier message de ce groupe:

```
usr.fileName=larmer
usr.toAgent=AGENT_VIOLET
```

4. Le moniteur est déclenché lorsque le groupe complet est écrit. L'agent remplace les propriétés de message IBM MQ dans le code XML de la tâche.

Cela entraîne la transformation du code XML de la tâche en:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_VENUS" QMgr="QM_VENUS"/>
    <destinationAgent agent="AGENT_VIOLET" QMgr="QM_MARS"/>
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <queue useGroups="true" groupId="{GROUPID}">START_QUEUE</queue>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/larmer.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Résultats

Le transfert défini par la tâche XML est effectué. Le groupe complet de messages lus à partir de la file d'attente START_QUEUE par AGENT_VENUS est écrit dans un fichier appelé /reports/larmer.rpt sur le système où s'exécute AGENT_VIOLET.

Que faire ensuite

Transfert de chaque message dans un fichier distinct

Si vous souhaitez surveiller une file d'attente et que chaque message soit transféré dans un fichier distinct, vous pouvez utiliser une technique similaire à celle décrite précédemment dans cette rubrique.

1. Créez le moniteur comme décrit précédemment, en spécifiant le paramètre **-tr completeGroups** dans la commande **fteCreateMonitor**.
2. Dans le code XML de la tâche, indiquez ce qui suit:

```
<queue useGroups="true" groupId="{GROUPID}">START_QUEUE</queue>
```

Toutefois, lorsque vous placez les messages dans la file d'attente source, ne les placez pas dans un groupe IBM MQ. Ajoutez des propriétés de message IBM MQ à chaque message. Par exemple, spécifiez la propriété `usr.filename` avec une valeur de nom de fichier unique pour chaque message. Ainsi, Managed File Transfer Agent traite chaque message de la file d'attente source comme un groupe distinct.

Exemple: erreur d'un transfert de message à fichier à l'aide des propriétés de message IBM MQ

Vous pouvez provoquer l'échec du transfert d'un message vers un fichier en définissant la propriété de message `usr.UserReturnCode` IBM MQ sur une valeur différente de zéro. Vous pouvez également spécifier des informations supplémentaires sur la raison de l'échec en définissant la propriété de message `usr.UserSupplement` IBM MQ.

Pourquoi et quand exécuter cette tâche

Dans cet exemple, un transfert est en cours entre la file d'attente INPUT_QUEUE et le fichier /home/user/output.file.

Un utilisateur crée des messages et les place dans la file d'attente INPUT_QUEUE. L'agent source consomme des messages de la file d'attente INPUT_QUEUE et envoie les données de transfert à l'agent cible. L'agent de destination écrit ces données dans le fichier /home/user/output.file.

Si un agent n'est pas répertorié par la commande **fteListAgents** ou n'est pas affiché dans le IBM MQ Explorer, utilisez le diagramme de diagnostic de la rubrique suivante pour localiser et résoudre le problème: Si votre agent n'est pas répertorié par la commande **fteListAgents**.

Arrêt d'un agent MFT

Vous pouvez arrêter un agent Managed File Transfer à partir de la ligne de commande. Lorsque vous arrêtez un agent, vous mettez l'agent au repos et vous l'autorisez à terminer son transfert de fichiers en cours avant de l'arrêter. Vous pouvez également spécifier le paramètre **-i** sur la ligne de commande pour arrêter immédiatement un agent. Une fois l'agent arrêté, vous ne pouvez pas l'utiliser pour transférer des fichiers tant que vous ne l'avez pas redémarré.

Avant de commencer

Si vous souhaitez vérifier les noms des agents associés à un gestionnaire de files d'attente, vous pouvez répertorier les agents à l'aide de la IBM MQ Explorer ou de la ligne de commande. Voir [Commande des agents fteList](#).

Pourquoi et quand exécuter cette tâche

Pour arrêter un agent à partir de la ligne de commande, voir [fteStopAgent](#).

Si vous avez configuré votre agent pour qu'il s'exécute en tant que service Windows, l'exécution de la commande **fteStopAgent** arrête également le service Windows. Vous pouvez également arrêter l'agent en arrêtant le service à l'aide de l'outil Windows Services. Pour plus d'informations, voir [«Démarrage d'un agent MFT en tant que service Windows»](#), à la page 159.

Arrêt d'un agent MFT sous z/OS

Si vous exécutez un Managed File Transfer Agent sous z/OS en tant que tâche démarrée à partir de JCL, l'agent accepte les commandes d'opérateur z/OS **MODIFY** et **STOP**, en plus de la commande **fteStopAgent**.

Une tâche démarrée est utilisée car elle s'exécute sous un ID utilisateur spécifique et n'est pas affectée par la déconnexion des utilisateurs.

Remarque :   Les tâches démarrées sont généralement exécutées sous un administrateur qui ne dispose peut-être pas de privilèges de connexion et il n'est donc pas possible de se connecter au système z/OS en tant qu'utilisateur sous lequel l'agent est exécuté. Les commandes **fteStartAgent**, **fteStopAgent**, **fteSetAgentTraceLevel** et **fteShowAgentDetails** avec le paramètre **-d** ne peuvent pas être émises pour cet agent.

Depuis IBM MQ 9.0.2 et IBM MQ 9.0.0 Fix Pack 1, la propriété d'agent **adminGroup** peut être utilisée avec les agents Managed File Transfer sous z/OS. Vous pouvez définir un groupe de gestionnaires de sécurité, par exemple MFTADMIN, puis ajouter l'ID utilisateur de la tâche démarrée et les ID TSO d'administrateur à ce groupe. Editez le fichier de propriétés de l'agent et définissez la propriété **adminGroup** pour qu'elle soit le nom de ce groupe de gestionnaires de sécurité.

```
adminGroup=MFTADMIN
```

Les membres de ce groupe peuvent ensuite émettre les commandes **fteStartAgent**, **fteStopAgent** et **fteSetAgentTraceLevel**, ainsi que la commande **fteShowAgentDetails** avec le paramètre **-d** spécifié, pour l'agent qui s'exécute en tant que tâche démarrée.

Pour plus d'informations, voir la propriété **adminGroup** dans [«Le fichier MFT agent.properties»](#), à la page 564.

Arrêt contrôlé de l'agent à l'aide de la commande z/OS MODIFY (F)

La commande **MODIFY** vous permet d'arrêter un agent de manière contrôlée comme alternative à la commande **fteStopAgent**. L'agent effectue tous les transferts en cours, mais il ne démarre aucun nouveau transfert.

Exemple :

```
F job_name,APPL=STOP
```

où *nom_travail* est le travail sous lequel le processus d'agent s'exécute.

Arrêt immédiat de l'agent à l'aide de la commande z/OS STOP (P)

La commande **STOP** équivaut à un arrêt immédiat en utilisant la commande **fteStopAgent** avec le paramètre **-i**. L'agent est arrêté immédiatement même s'il est en train de transférer un fichier.

Exemple :

```
P job_name
```

où *nom_travail* est le travail sous lequel le processus d'agent s'exécute.

Référence associée

«Démarrage d'un agent MFT sous z/OS», à la page 162

Sous z/OS, en plus d'exécuter la commande **fteStartAgent** à partir d'une session UNIX System Services, vous pouvez démarrer un agent en tant que tâche démarrée à partir de JCL sans avoir besoin d'une session interactive.

«Le fichier MFT agent.properties», à la page 564

Chaque Managed File Transfer Agent possède son propre fichier de propriétés, `agent.properties`, qui doit contenir les informations qu'un agent utilise pour se connecter à son gestionnaire de files d'attente. Le fichier `agent.properties` peut également contenir des propriétés qui modifient le comportement de l'agent.

Le pont de protocole

Le pont de protocole permet à votre réseau Managed File Transfer (MFT) d'accéder aux fichiers stockés sur un serveur de fichiers en dehors de votre réseau MFT, dans votre domaine local ou à un emplacement distant. Ce serveur de fichiers peut utiliser les protocoles de réseau FTP, FTPS ou SFTP. Chaque serveur de fichiers a besoin d'au moins un agent dédié. L'agent dédié est appelé agent de pont de protocole. Un agent de pont peut interagir avec plusieurs serveurs de fichiers.

Le pont de protocole est disponible dans le cadre du composant Service de Managed File Transfer. Vous pouvez avoir plusieurs agents dédiés sur un seul système exécutant MFT qui se connectent à différents serveurs de fichiers.

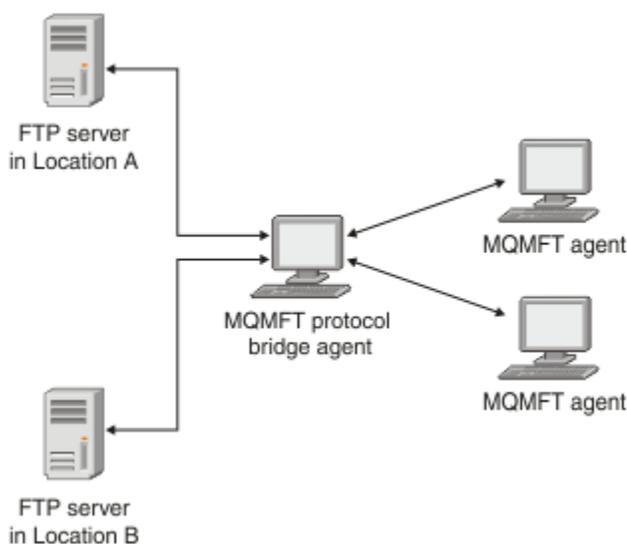
Vous pouvez utiliser un agent de pont de protocole pour transférer des fichiers vers plusieurs nœuds finaux simultanément. MFT fournit un fichier appelé `ProtocolBridgeProperties.xml` que vous pouvez éditer pour définir les différents serveurs de fichiers de protocole vers lesquels vous souhaitez transférer des fichiers. La commande **fteCreateBridgeAgent** ajoute les détails du serveur de fichiers de protocole par défaut à `ProtocolBridgeProperties.xml` pour vous. Ce fichier est décrit dans «Format du fichier de propriétés du pont de protocole», à la page 592.

Vous pouvez utiliser l'agent de pont de protocole pour effectuer les actions suivantes:

- Téléchargez des fichiers depuis le réseau MFT vers un serveur distant à l'aide de FTP, FTPS ou SFTP.
- Téléchargement de fichiers depuis un serveur distant, à l'aide de FTP, FTPS ou SFTP, sur le réseau MFT

Remarque : L'agent de pont de protocole peut prendre en charge uniquement les serveurs FTP, FTPS ou SFTP qui permettent d'accéder aux fichiers par leur chemin d'accès absolu. Si un chemin de fichier relatif est spécifié dans une demande de transfert, l'agent de pont de protocole tente de convertir le

chemin relatif en chemin de fichier absolu en fonction du répertoire de base utilisé pour la connexion au serveur de protocole. Les serveurs de protocole qui autorisent l'accès aux fichiers basés uniquement sur le répertoire en cours ne sont pas pris en charge par l'agent de pont de protocole.



Le diagramme montre deux serveurs FTP, à des emplacements différents. Les serveurs FTP sont utilisés pour échanger des fichiers avec les agents Managed File Transfer. L'agent de pont de protocole se trouve entre les serveurs FTP et le reste du réseau MFT et est configuré pour communiquer avec les deux serveurs FTP.

Vérifiez que vous disposez d'un autre agent dans votre réseau MFT en plus de l'agent de pont de protocole. L'agent de pont de protocole est un pont vers le serveur FTP, FTPS ou SFTP uniquement et n'écrit pas les fichiers transférés sur le disque local. Si vous souhaitez transférer des fichiers vers ou depuis le serveur FTP, FTPS ou SFTP, vous devez utiliser l'agent de pont de protocole comme destination ou source du transfert de fichiers (représentant le serveur FTP, FTPS ou SFTP) et un autre agent standard comme source ou destination correspondante.

Lorsque vous transférez des fichiers à l'aide du pont de protocole, le pont doit avoir le droit de lire le répertoire source ou de destination contenant les fichiers que vous souhaitez transférer. Par exemple, si vous souhaitez transférer des fichiers à partir du répertoire `/home/fte/bridge` qui dispose de droits d'exécution (`d -- x -- x -- x`) uniquement, les transferts que vous tentez d'effectuer à partir de ce répertoire échouent avec le message d'erreur suivant:

```
BFGBR0032E: Attempt to read filename from the protocol file server  
has failed with server error 550. Failed to open file.
```

Configuration d'un agent de pont de protocole

Un agent de pont de protocole est similaire à un agent MFT standard. Créez un agent de pont de protocole à l'aide de la commande **fteCreateBridgeAgent**. Vous pouvez configurer un agent de pont de protocole à l'aide du fichier `ProtocolBridgeProperties.xml`, qui est décrit dans «[Format du fichier de propriétés du pont de protocole](#)», à la page 592. Si vous utilisez une version antérieure, configurez l'agent à l'aide des propriétés de pont de protocole spécifiques décrites dans [Propriétés d'agent avancées](#). Pour toutes les versions, vous pouvez également configurer un mappage de données d'identification comme décrit dans «[Mappage des données d'identification pour un serveur de fichiers](#)», à la page 228. Une fois que vous avez configuré un agent de pont de protocole pour un serveur de fichiers de protocole particulier, vous ne pouvez utiliser cet agent qu'à cette fin.

Reprise du pont de protocole

Si l'agent de pont de protocole ne parvient pas à se connecter au serveur de fichiers car ce dernier n'est pas disponible, toutes les demandes de transfert de fichiers sont mises en file d'attente jusqu'à ce que

le serveur de fichiers soit disponible. Si l'agent de pont de protocole ne parvient pas à se connecter au serveur de fichiers car l'agent utilise des données d'identification incorrectes, le transfert échoue et le message du journal de transfert reflète cette erreur. Si l'agent de pont de protocole est arrêté pour une raison quelconque, tous les transferts de fichiers demandés sont conservés et continuent lorsque le pont de protocole est redémarré.

Lors du transfert de fichiers, les fichiers sont généralement écrits en tant que fichiers temporaires sur la destination et sont ensuite renommés lorsque le transfert est terminé. Toutefois, si la destination de transfert est un serveur de fichiers de protocole configuré en écriture limitée (les utilisateurs peuvent télécharger des fichiers sur le serveur de fichiers de protocole mais ne peuvent pas modifier ces fichiers téléchargés de quelque manière que ce soit ; les utilisateurs peuvent écrire une seule fois), les fichiers transférés sont écrits directement sur la destination. Cela signifie que si un problème se produit lors du transfert, les fichiers partiellement écrits restent sur le serveur de fichiers de protocole de destination et Managed File Transfer ne peut pas supprimer ou éditer ces fichiers. Dans ce cas, le transfert échoue.

Définition des propriétés des serveurs de fichiers de protocole à l'aide du fichier `ProtocolBridgeProperties.xml`

Définissez les propriétés d'un ou de plusieurs serveurs de fichiers de protocole vers et depuis lesquels vous souhaitez transférer des fichiers à l'aide du fichier `ProtocolBridgeProperties.xml`, fourni par Managed File Transfer dans le répertoire de configuration de l'agent.

Pourquoi et quand exécuter cette tâche

La commande `fteCreateBridgeAgent` crée le fichier `ProtocolBridgeProperties.xml` dans le répertoire de configuration de l'agent `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name`. La commande crée également une entrée dans le fichier pour le serveur de fichiers de protocole par défaut, si une valeur par défaut a été spécifiée lors de l'exécution de la commande.

Le message BFGCL0392I indique l'emplacement du fichier `ProtocolBridgeProperties.xml`.

```
<?xml version="1.0" encoding="IBM-1047"?>
<!--
This ProtocolBridgeProperties.xml file determines the protocol servers that will be accessed by
the
MQMFT protocol bridge agent.

Each protocol server is defined using either a <tns:ftpServer>, <tns:ftpsServer>, or
<tns:sftpServer>
element - depending on the protocol used to communicate with the server. When the protocol
bridge agent participates in a managed file transfer it will determine which server to used
based on
the prefix (if any) present on the file path. For example a file path of 'server1:/home/user/
file.txt' would
be interpreted as a request to transfer /home/user/file.txt using 'server1'. The server name
is compared
to the 'name' attribute of each <tns:ftpServer>, <tns:ftpsServer> or <tns:sftpServer> element
in this
XML document and the first match is used to determine which protocol server the protocol bridge
agent will connect to. If no match is found then the managed file transfer operation will fail.

If a file path is not prefixed with a server name, for example '/home/user/file.txt' then this
XML
document can specify a default server to use for the managed file transfer. To specify a
default server use the <tns:defaultServer> element as the first element inside the
<tns:serverProperties>
element. The default server will be used whenever the protocol bridge agent participates in
a managed file transfer for file names which do not specify a prefix.

An optional <tns:limits> element can be specified within each server definition. This element
contains
attributes that govern the amount of resources used by each defined server.

An optional <tns:credentialsFile> element can be specified within each serverProperties
definition. This
element contains a path to a file containing credentials to be used when connecting to defined
servers.
```

An example `ProtocolBridgeProperties.xml` file is as follows:

```

<?xml version="1.0" encoding="UTF-8"?>
<tns:serverProperties xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeProperties"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeProperties
ProtocolBridgeProperties.xsd">

  <tns:credentialsFile path="$HOME/ProtocolBridgeCredentials.xml"/>

  <tns:defaultServer name="myFTPserver"/>

  <tns:ftpServer name="myFTPserver" host="windows.hursley.ibm.com" port="1234"
platform="windows"
  timeZone="Europe/London" locale="en_GB" fileEncoding="UTF-8"
  listFormat="unix" limitedWrite="false">

    <tns:limits maxListFileNames="100" maxListDirectoryLevels="999999999"
      maxReconnectRetry="2" reconnectWaitPeriod="10"
      maxSessions="60" socketTimeout="30"/>

  </tns:ftpServer>

  <tns:ftpsServer name="myFTPSserver" host="unix.hursley.ibm.com" platform="unix"
  timeZone="Europe/London" locale="en_GB" fileEncoding="UTF8"
  listFormat="unix" limitedWrite="false" ftpsType="explicit"
  trustStore="C:\FTE\keystores\myFTPSserver\FTPSKeyStore.jks"
trustStorePassword="password">

    <tns:limits maxReconnectRetry="10" connectionTimeout="10"/>

  </tns:ftpsServer>

  <tns:sftpServer name="mySFTPserver" host="windows.hursley.ibm.com" platform="windows"
  timeZone="Europe/London" locale="en_GB" fileEncoding="UTF-8"
  limitedWrite="false">

    <tns:limits connectionTimeout="60"/>

  </tns:sftpServer>
</tns:serverProperties>

```

This example shows the outermost <tns:serverProperties> element which must exist for the document to be valid, an optional <tns:defaultServer> element, as well as definitions for an FTP, FTPS and SFTP server.

The attributes of the <tns:ftpServer>, <tns:ftpsServer> and <tns:sftpServer> elements determine the characteristics of the connection established to the server. These attributes correspond to the command line parameters for the 'fteCreateBridgeAgent' command.

The following attributes are valid for all of the <tns:ftpServer>, <tns:ftpsServer> and <tns:sftpServer> elements: name, host, port, platform, fileEncoding, limitedWrite and controlEncoding.

The following attributes are valid for the <tns:ftpServer> and <tns:ftpsServer> elements: timeZone, locale, listFormat, listFileRecentDateFormat, listFileOldDateFormat, and monthShortNames.

The following attributes are valid for the <tns:ftpServer> element only: passiveMode

The following attributes are valid for the <tns:ftpsServer> element only: ftpsType, trustStore, trustStorePassword, trustStoreType, keyStore, keyStorePassword, keyStoreType, ccc, protFirst, auth, and connectTimeout.

The following attributes are valid for the <tns:limits> element within all of the <tns:ftpServer>, <tns:ftpsServer> and <tns:sftpServer> elements: maxListFileNames, maxListDirectoryLevels, maxReconnectRetry, reconnectWaitPeriod, maxSessions and socketTimeout

```

-->
<tns:serverProperties xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeProperties"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeProperties
ProtocolBridgeProperties.xsd">

```

<!-- By default the location of the credentials file is in the home directory of the user that started the -->

```

    <!-- protocol bridge agent. If you wish to specify a different location use the
credentialsFile element to -->
    <!-- do this. For
example:
    <!-- <tns:credentialsFile path="/test/
ProtocolBridgeCredentials.xml"/>
-->

    <tns:defaultServer name="WINMVSCA.HURSLEY.IBM.COM"/>
    <tns:ftpServer name="WINMVSCA.HURSLEY.IBM.COM" host="WINMVSCA.HURSLEY.IBM.COM"
platform="UNIX"
        timeZone="Europe/London" locale="en-GB" fileEncoding="US-ASCII"
        listFormat="unix" limitedWrite="false" />

    <!-- Define servers here -->
</tns:serverProperties>

```

La commande peut générer le message suivant: BFGCL0532I:

Cet agent ne peut fonctionner que si un fichier de données d'identification supplémentaire est créé manuellement. Par défaut, ce fichier est appelé ProtocolBridgeCredentials.xml et se trouve dans le répertoire principal du répertoire de l'utilisateur qui démarre l'agent. Par exemple, si cet utilisateur a démarré l'agent l'emplacement serait: \$HOME/ProtocolBridgeCredentials.xml

Si vous utilisez un fichier de données d'identification:

1. Pour plus d'informations sur la façon d'en créer un, voir le texte suivant.
2. Le fichier de données d'identification doit se trouver dans un répertoire avec des droits d'accès restreints. Par exemple, il ne doit pas y avoir d'accès en lecture pour les autres utilisateurs.
3. Indiquez l'emplacement du répertoire du fichier de données d'identification dans la variable d'environnement \$HOME pour l'ID utilisateur de l'agent démarré, ou éditez le fichier ProtocolBridgeProperties.xml et indiquez l'emplacement dans:

```
<tns:credentialsFile path="/test/ProtocolBridgeCredentials.xml"/>
```

Si vous souhaitez ajouter d'autres serveurs de protocole autres que ceux par défaut, éditez ce fichier pour définir leurs propriétés. Cet exemple ajoute un serveur FTP supplémentaire.

Remarque : L'agent de pont de protocole ne prend pas en charge le verrouillage de fichier. En effet, Managed File Transfer ne prend pas en charge le mécanisme de verrouillage de fichier sur un serveur de fichiers.

Procédure

1. Définissez un serveur de fichiers de protocole en insérant les lignes suivantes dans le fichier en tant qu'élément enfant de <tns:serverProperties>:

```

<tns:ftpServer name="myserver" host="myhost.hursley.ibm.com" port="1234"
platform="windows"
        timeZone="Europe/London" locale="en-GB" fileEncoding="UTF-8"
        listFormat="unix" limitedWrite="false" >
<tns:limits maxListFileNames="10" maxListDirectoryLevels="500"/>

```

2. Modifiez ensuite la valeur des attributs:

- name est le nom de votre serveur de fichiers de protocole
- host est le nom d'hôte ou l'adresse IP du serveur de fichiers de protocole
- port est le numéro de port du serveur de fichiers de protocole
- platform est la plateforme sur laquelle s'exécute le serveur de fichiers de protocole
- timeZone est le fuseau horaire dans lequel s'exécute le serveur de fichiers de protocole
- locale est la langue utilisée sur le serveur de fichiers de protocole
- fileEncoding est le codage de caractères du serveur de fichiers de protocole
- listFormat est le format de liste de fichiers renvoyé par le serveur de fichiers de protocole

- `limitedWrite` détermine s'il convient de suivre le mode par défaut lors de l'écriture sur un serveur de fichiers, qui consiste à créer un fichier temporaire, puis à renommer ce fichier lorsque le transfert est terminé. Pour un serveur de fichiers configuré en écriture seule, le fichier est créé directement avec son nom final. La valeur de cette propriété peut être `true` ou `false`. L'attribut `limitedWrite` et la propriété d'agent `doNotUseTempOutputFile` sont utilisés ensemble dans le cas des agents de pont de protocole. Si vous souhaitez utiliser des fichiers temporaires, vous ne devez pas définir la valeur de `doNotUseTempOutputFile` et vous devez définir la valeur de `limitedWrite` sur `false`. Toute autre combinaison de paramètres signifie que les fichiers temporaires ne seront pas utilisés.
- `maxListFileNames` est le nombre maximal de noms collectés lors de l'analyse d'un répertoire sur le serveur de fichiers de protocole à la recherche de noms de fichier.
- `maxListDirectoryLevels` est le nombre maximal de niveaux de répertoire à récursivité lors de l'analyse d'un répertoire sur le serveur de fichiers de protocole à la recherche de noms de fichier.

Pour plus de détails sur ces attributs, notamment s'ils sont obligatoires ou facultatifs et leurs valeurs par défaut, voir [«Format du fichier de propriétés du pont de protocole»](#), à la page 592.

Référence associée

[«Format du fichier de propriétés du pont de protocole»](#), à la page 592

Le fichier `ProtocolBridgeProperties.xml` du répertoire de configuration de l'agent définit les propriétés des serveurs de fichiers de protocole.

[«Expressions régulières utilisées par MFT»](#), à la page 679

Managed File Transfer utilise des expressions régulières dans un certain nombre de scénarios. Par exemple, des expressions régulières sont utilisées pour faire correspondre des ID utilisateur pour les données d'identification de sécurité `Connect:Direct` ou pour fractionner un fichier en plusieurs messages en créant un nouveau message chaque fois qu'une expression régulière est mise en correspondance. La syntaxe d'expression régulière utilisée par Managed File Transfer est la syntaxe prise en charge par l'API `java.util.regex`. Cette syntaxe d'expression régulière est similaire, mais pas identique, à la syntaxe d'expression régulière utilisée par le langage Perl.

ProtocolBridgePropertiesExit2: recherche des propriétés du serveur de fichiers de protocole

Si vous disposez d'un grand nombre de serveurs de fichiers de protocole, vous pouvez implémenter l'interface `com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit2` pour rechercher les propriétés de serveur de fichiers de protocole référencées dans les transferts. Vous pouvez implémenter cette interface de préférence à la gestion d'un fichier `ProtocolBridgeProperties.xml`. Managed File Transfer fournit un exemple d'exit utilisateur qui recherche les propriétés du serveur de fichiers de protocole.

Configuration des exits utilisateur qui recherchent les propriétés de pont de protocole

Pourquoi et quand exécuter cette tâche

Tout exit utilisateur qui recherche des propriétés de pont de protocole doit implémenter l'interface `com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit2`. Pour plus d'informations, voir la section [«Interface ProtocolBridgePropertiesExit2.java»](#), à la page 934.

Vous pouvez chaîner plusieurs exits de propriétés de serveur de protocole de la même manière que d'autres exits utilisateur. Les exits sont appelés dans l'ordre dans lequel ils sont spécifiés à l'aide de la propriété `protocolBridgePropertiesExitClasses` dans le fichier de propriétés de l'agent. Les méthodes d'initialisation sont toutes renvoyées séparément et si une ou plusieurs d'entre elles renvoient la valeur `false`, l'agent ne démarre pas. L'erreur est signalée dans le journal des événements de l'agent.

Un seul résultat global est renvoyé pour les méthodes `getProtocolServerProperties` de tous les exits. Si la méthode renvoie un objet de propriétés comme code de résultat, cette valeur correspond au résultat renvoyé et les méthodes `getProtocolServerProperties` des exits suivants ne sont pas appelées. Si la méthode renvoie une valeur `null` comme code de résultat, la méthode `getProtocolServerProperties` de

l'exit suivant est appelée. S'il n'y a pas d'exit suivant, le résultat null est renvoyé. Un code de résultat global nul est considéré comme un échec de recherche par l'agent de pont de protocole.

Il est recommandé d'utiliser l'interface `ProtocolBridgePropertiesExit2.java`, mais pour plus d'informations sur l'interface `ProtocolBridgePropertiesExit.java`, voir «[ProtocolBridgePropertiesExit: recherche des propriétés du serveur de fichiers de protocole](#)», à la page 226.

Pour exécuter votre exit, procédez comme suit:

Procédure

1. Compilez l'exit utilisateur des propriétés du serveur de protocole.
2. Créez un fichier d'archive Java (JAR) contenant l'exit compilé et sa structure de package.
3. Placez le fichier JAR contenant la classe d'exit dans le répertoire `exits` de l'agent de pont de protocole. Ce répertoire se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name`.
4. Editez le fichier de propriétés de l'agent de pont de protocole pour inclure la propriété `protocolBridgePropertiesExitClasses`. Pour la valeur de cette propriété, spécifiez une liste de classes séparées par des virgules qui implémentent un exit utilisateur de propriétés de serveur de pont de protocole. Les classes d'exit sont appelées dans l'ordre dans lequel elles sont spécifiées dans cette liste. Pour plus d'informations, voir la section «[Le fichier MFT agent.properties](#)», à la page 564.
5. Vous pouvez éventuellement spécifier la propriété `protocolBridgePropertiesConfiguration`. La valeur que vous spécifiez pour cette propriété est transmise en tant que chaîne à la méthode `initialize()` des classes d'exit spécifiées par `protocolBridgePropertiesExitClasses`. Pour plus d'informations, voir «[Le fichier MFT agent.properties](#)», à la page 564.

Utilisation de l'exemple d'exit utilisateur

Pourquoi et quand exécuter cette tâche

Un exemple d'exit utilisateur qui recherche les propriétés de pont de protocole est fourni dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/protocolBridge` et dans la rubrique «[Exemple d'exit utilisateur de propriétés de pont de protocole](#)», à la page 273.

L'exit `SamplePropertiesExit2.java` lit un fichier de propriétés qui contient les propriétés des serveurs de protocole. Le format de chaque entrée du fichier de propriétés est le suivant:

```
serverName=type://host:port
```

L'emplacement du fichier de propriétés est extrait de la propriété d'agent de pont de protocole `protocolBridgePropertiesConfiguration`.

Pour exécuter l'exemple d'exit utilisateur, procédez comme suit:

Procédure

1. Compilez le fichier `SamplePropertiesExit2.java`.
2. Créez un fichier JAR contenant l'exit compilé et sa structure de package.
3. Placez le fichier JAR dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent/exits`.
4. Editez le fichier `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name/agent.properties` pour qu'il contienne la ligne suivante:

```
protocolBridgePropertiesExitClasses=SamplePropertiesExit2
```

5. Créez un fichier de propriétés de pont de protocole, par exemple `protocol_bridge_properties.properties`, dans le répertoire `MQ_DATA_PATH/mqft/config/`

`coordination_queue_manager/agents/bridge_agent`. Modifiez ce fichier pour inclure des entrées au format suivant:

```
serverName=type://host:port
```

6. Editez le fichier `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent/agent.properties` pour qu'il contienne la ligne suivante:

```
protocolBridgePropertiesConfiguration=MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent/protocol_bridge_properties.properties
```

Vous devez utiliser le chemin d'accès absolu au fichier `protocol_bridge_properties.properties`.

7. Démarrez l'agent de pont de protocole à l'aide de la commande **fteStartAgent**.

Concepts associés

«Le pont de protocole», à la page 219

Le pont de protocole permet à votre réseau Managed File Transfer (MFT) d'accéder aux fichiers stockés sur un serveur de fichiers en dehors de votre réseau MFT, dans votre domaine local ou à un emplacement distant. Ce serveur de fichiers peut utiliser les protocoles de réseau FTP, FTPS ou SFTP. Chaque serveur de fichiers a besoin d'au moins un agent dédié. L'agent dédié est appelé agent de pont de protocole. Un agent de pont peut interagir avec plusieurs serveurs de fichiers.

Référence associée

«Interface `ProtocolBridgePropertiesExit2.java`», à la page 934

«Exemple d'exit utilisateur de propriétés de pont de protocole», à la page 273

«Le fichier MFT `agent.properties`», à la page 564

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.

«`fteCreateBridgeAgent` (créer et configurer un agent de pont de protocole MFT)», à la page 403

La commande **fteCreateBridgeAgent** crée un agent de pont de protocole Managed File Transfer et sa configuration associée. Créez un agent de pont de protocole pour chaque serveur de fichiers auquel vous souhaitez envoyer des fichiers et duquel vous souhaitez les recevoir.

ProtocolBridgePropertiesExit: recherche des propriétés du serveur de fichiers de protocole

Si vous disposez d'un grand nombre de serveurs de fichiers de protocole, vous pouvez implémenter l'interface `com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit` pour rechercher les propriétés de serveur de fichiers de protocole référencées dans les transferts. Vous pouvez implémenter cette interface de préférence à la gestion d'un fichier `ProtocolBridgeProperties.xml`. Il est recommandé d'utiliser l'interface `ProtocolBridgePropertiesExit2.java`, mais l'interface `ProtocolBridgePropertiesExit.java` est également prise en charge. Si vous disposez d'une implémentation existante de l'interface `ProtocolBridgePropertiesExit.java` à partir de IBM WebSphere MQ File Transfer Edition, vous pouvez l'utiliser dans IBM WebSphere MQ 7.5 ou version ultérieure. La nouvelle méthode d'emplacement `getCredential` dans `ProtocolBridgePropertiesExit2.java` utilise l'emplacement par défaut du fichier `ProtocolBridgeCredentials.xml`, qui correspond à votre répertoire de base.

Configuration des exits utilisateur qui recherchent les propriétés de pont de protocole

Tout exit utilisateur qui recherche des propriétés de pont de protocole doit implémenter l'interface `com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit`. Pour plus d'informations, voir la section «Interface `ProtocolBridgePropertiesExit.java`», à la page 227.

Vous pouvez chaîner plusieurs exits de propriétés de serveur de protocole de la même manière que d'autres exits utilisateur. Les exits sont appelés dans l'ordre dans lequel ils sont spécifiés à l'aide de la propriété `protocolBridgePropertiesExitClasses` dans le fichier de propriétés de l'agent. Les méthodes d'initialisation sont toutes renvoyées séparément et si une ou plusieurs d'entre elles renvoient la valeur `false`, l'agent ne démarre pas. L'erreur est signalée dans le journal des événements de l'agent.

Un seul résultat global est renvoyé pour les méthodes `getProtocolServerProperties` de tous les exits. Si la méthode renvoie un objet de propriétés comme code de résultat, cette valeur correspond au résultat renvoyé et les méthodes `getProtocolServerProperties` des exits suivants ne sont pas appelées. Si la méthode renvoie une valeur `null` comme code de résultat, la méthode `getProtocolServerProperties` de l'exit suivant est appelée. S'il n'y a pas d'exit suivant, le résultat `null` est renvoyé. Un code de résultat global nul est considéré comme un échec de recherche par l'agent de pont de protocole.

Pour exécuter votre exit, procédez comme suit:

1. Compilez l'exit utilisateur des propriétés du serveur de protocole.
2. Créez un fichier d'archive Java (JAR) contenant l'exit compilé et sa structure de package.
3. Placez le fichier JAR contenant la classe d'exit dans le répertoire `exits` de l'agent de pont de protocole. Ce répertoire se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name`.
4. Editez le fichier de propriétés de l'agent de pont de protocole pour inclure la propriété `protocolBridgePropertiesExitClasses`. Pour la valeur de cette propriété, spécifiez une liste de classes séparées par des virgules qui implémentent un exit utilisateur de propriétés de serveur de pont de protocole. Les classes d'exit sont appelées dans l'ordre dans lequel elles sont spécifiées dans cette liste. Pour plus d'informations, voir la section «Le fichier `MFT agent.properties`», à la page 564.
5. Vous pouvez éventuellement spécifier la propriété `protocolBridgePropertiesConfiguration`. La valeur que vous spécifiez pour cette propriété est transmise en tant que chaîne à la méthode `initialize()` des classes d'exit spécifiées par `protocolBridgePropertiesExitClasses`. Pour plus d'informations, voir «Le fichier `MFT agent.properties`», à la page 564.

Interface `ProtocolBridgePropertiesExit.java`

```
package com.ibm.wmqfte.exitroutine.api;

import java.util.Map;
import java.util.Properties;

/**
 * An interface that is implemented by classes that are to be invoked as part of
 * user exit routine processing. This interface defines methods that will be
 * invoked by a protocol bridge agent to look up properties for protocol servers
 * that are referenced in transfers.
 * <p>
 * There will be one instance of each implementation class for each protocol
 * bridge agent. The methods can be called from different threads so the methods
 * must be synchronised.
 */
public interface ProtocolBridgePropertiesExit {

    /**
     * Invoked once when a protocol bridge agent is started. It is intended to
     * initialize any resources that are required by the exit.
     *
     * @param bridgeProperties
     *        The values of properties defined for the protocol bridge.
     *        These values can only be read, they cannot be updated by the
     *        implementation.
     * @return {@code true} if the initialization is successful and {@code
     *         false} if unsuccessful. If {@code false} is returned from an exit
     *         the protocol bridge agent will not start.
     */
    public boolean initialize(final Map<String, String> bridgeProperties);

    /**
     * Obtains a set of properties for the specified protocol server name.
     * <p>
     * The returned {@link Properties} must contain entries with key names

```

```

* corresponding to the constants defined in
* {@link ProtocolServerPropertyConstants} and in particular must include an
* entry for all appropriate constants described as required.
*
* @param protocolServerName
*       The name of the protocol server whose properties are to be
*       returned. If a null or a blank value is specified, properties
*       for the default protocol server are to be returned.
* @return The {@link Properties} for the specified protocol server, or null
*         if the server cannot be found.
*/
public Properties getProtocolServerProperties(
    final String protocolServerName);

/**
* Invoked once when a protocol bridge agent is shut down. It is intended to
* release any resources that were allocated by the exit.
*
* @param bridgeProperties
*       The values of properties defined for the protocol bridge.
*       These values can only be read, they cannot be updated by the
*       implementation.
*/
public void shutdown(final Map<String, String> bridgeProperties);
}

```

Mapper les données d'identification pour un serveur de fichiers

Mappez les données d'identification de l'utilisateur dans Managed File Transfer aux données d'identification de l'utilisateur sur le serveur de fichiers à l'aide de la fonction de mappage des données d'identification par défaut de l'agent de pont de protocole 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.

Mapper les données d'identification pour un serveur de fichiers à l'aide du fichier ProtocolBridgeCredentials.xml

Mappez les données d'identification de l'utilisateur dans Managed File Transfer aux données d'identification de l'utilisateur sur le serveur de fichiers à l'aide de la fonction de mappage des données d'identification par défaut de l'agent de pont de protocole. Managed File Transfer fournit un fichier XML que vous pouvez éditer pour inclure vos données d'identification.

Pourquoi et quand exécuter cette tâche

Le fichier ProtocolBridgeCredentials.xml doit être créé manuellement par l'utilisateur. Par défaut, l'emplacement de ce fichier est le répertoire de base de l'utilisateur qui a démarré l'agent de pont de protocole, mais il peut être stocké n'importe où sur le système de fichiers accessible par l'agent. Pour spécifier un autre emplacement, ajoutez l'élément <credentialsFile> au fichier ProtocolBridgeProperties.xml. Par exemple :

```
<tns:credentialsFile path="/example/path/to/ProtocolBridgeCredentials.xml"/>
```

Avant de pouvoir utiliser un agent de pont de protocole, configurez le mappage des données d'identification en éditant ce fichier pour inclure les informations d'hôte, d'utilisateur et de données d'identification. Pour plus d'informations et pour obtenir des exemples, voir [«Format de fichier des données d'identification du pont de protocole»](#), à la page 589.

 Lorsque vous créez le fichier ProtocolBridgeCredentials.xml sur une plateforme z/OS à l'aide de IBM WebSphere MQ 7.5 ou d'une version antérieure, vous devez définir une balise de fichier avant d'éditer le fichier. Exécutez la commande suivante pour marquer le fichier comme ayant un contenu ASCII:

```
chtag -t -c IS08859-1 ProtocolBridgeCredentials.xml
```

Remarque :  Sous z/OS, vous pouvez stocker le fichier de données d'identification du pont de protocole dans un fichier, où le nom du fichier .xml peut être spécifié par l'utilisateur.

Procédure

1. Editez la ligne `<tns:server name="server name">` pour remplacer la valeur de l'attribut de nom par le nom du serveur dans le fichier `ProtocolBridgeProperties.xml`.

Les agents de pont de protocole créés pour IBM WebSphere MQ File Transfer Edition 7.0.4 et les versions antérieures ne possèdent pas de fichier `ProtocolBridgeProperties.xml` (ou d'exits utilisateur associés). Par conséquent, pour IBM WebSphere MQ File Transfer Edition 7.0.4 Fix Pack 1 et les versions ultérieures, le nom de serveur est automatiquement affecté au nom d'hôte du serveur. Par conséquent, si vous utilisez un fichier `ProtocolBridgeCredentials.xml` mis à jour avec des entrées `<server>`, un nom correspondant au nom d'hôte du serveur correspondra.

Vous pouvez utiliser l'attribut de modèle pour indiquer que vous avez utilisé un nom de serveur contenant des caractères génériques ou des expressions régulières. Par exemple :

```
<tns:server name="serverA*" pattern="wildcard">
```

2. Insérez l'ID utilisateur et les données d'identification dans le fichier en tant qu'éléments enfant de `<tns:server>`.

Vous pouvez insérer un ou plusieurs des éléments suivants dans le fichier:

- Si le serveur de fichiers de protocole est un serveur FTP, FTPS ou SFTP, vous pouvez utiliser des mots de passe pour authentifier l'utilisateur qui demande le transfert. Insérez les lignes suivantes dans le fichier:

```
<tns:user name="FTE User ID"
  serverUserId="Server User ID"
  serverPassword="Server Password">
</tns:user>
```

Modifiez ensuite la valeur des attributs.

- `name` est une expression régulière Java qui correspond à l'ID utilisateur MQMD associé à la demande de transfert MFT
- `serverUserId` est la valeur qui est transmise au serveur de fichiers de protocole en tant qu'ID utilisateur de connexion. Si l'attribut `serverUserId` n'est pas spécifié, l'ID utilisateur MQMD associé à la demande de transfert MFT est utilisé à la place.
- `serverPassword` est le mot de passe associé à `serverUserId`.

L'attribut `name` peut contenir une expression régulière Java. Le programme de mappage de données d'identification tente de faire correspondre l'ID utilisateur MQMD de la demande de transfert MFT à cette expression régulière. L'agent de pont de protocole tente de faire correspondre l'ID utilisateur MQMD à l'expression régulière dans l'attribut de nom des éléments `<tns:user>` dans l'ordre dans lequel les éléments existent dans le fichier. Lorsqu'une correspondance est trouvée, l'agent de pont de protocole ne recherche pas d'autres correspondances. Si une correspondance est trouvée, les valeurs `serverUserId` et `serverPassword` correspondantes sont transmises au serveur de fichiers de protocole en tant qu'ID utilisateur et mot de passe de connexion. Les correspondances d'ID utilisateur MQMD sont sensibles à la casse.

- Si le serveur de fichiers de protocole est un serveur SFTP, vous pouvez utiliser des clés publiques et privées pour authentifier l'utilisateur qui demande le transfert. Insérez les lignes suivantes dans le fichier et modifiez la valeur des attributs. L'élément `<tns:user>` peut contenir un ou plusieurs éléments `<tns:privateKey>`.

```
<tns:user name="FTE User ID"
  serverUserId="Server User ID"
  hostKey="Host Key">
  <tns:privateKey associationName="association"
```

```
keyPassword="Private key password">
  Private key file text
</tns:privateKey>
</tns:user>
```

- name est une expression régulière Java qui correspond à l'ID utilisateur MQMD associé à la demande de transfert MFT
- serverUserId est la valeur qui est transmise au serveur de fichiers de protocole en tant qu'ID utilisateur de connexion. Si l'attribut serverUserId n'est pas spécifié, l'ID utilisateur MQMD associé à la demande de transfert MFT est utilisé à la place.
- hostKey est la clé attendue renvoyée par le serveur lors de la connexion
- key est la clé privée de serverUserId
- keyPassword est le mot de passe de la clé permettant de générer des clés publiques
- associationName est une valeur utilisée pour identifier à des fins de trace et de consignation

L'attribut name peut contenir une expression régulière Java . Le programme de mappage de données d'identification tente de faire correspondre l'ID utilisateur MQMD de la demande de transfert MFT à cette expression régulière. L'agent de pont de protocole tente de faire correspondre l'ID utilisateur MQMD à l'expression régulière dans l'attribut de nom des éléments < tns: user> dans l'ordre dans lequel les éléments existent dans le fichier. Lorsqu'une correspondance est trouvée, l'agent de pont de protocole ne recherche pas d'autres correspondances. Si une correspondance est trouvée, les valeurs serverUserId et key correspondantes sont utilisées pour authentifier l'utilisateur MFT auprès du serveur de fichiers de protocole. Les correspondances d'ID utilisateur MQMD sont sensibles à la casse.

Pour plus d'informations sur l'utilisation de clés privées avec un agent de pont de protocole, voir [«Exemple: Comment configurer un agent de pont de protocole pour utiliser des données d'identification de clé privée avec un serveur SFTP UNIX»](#), à la page 233.

Remarque :  z/OS

Lorsque la demande de transfert est écrite dans la file d'attente de commandes, l'ID utilisateur MQMD peut être converti en majuscules si la file d'attente de commandes de l'agent source se trouve sur un système z/OS ou IBM i . Par conséquent, l'ID utilisateur MQMD du même utilisateur d'origine peut arriver à l'exit de données d'identification dans la casse d'origine ou être converti en majuscules en fonction de l'agent source spécifié dans la demande de transfert. L'exit de mappage des données d'identification par défaut effectue des correspondances sensibles à la casse avec l'ID utilisateur MQMD fourni, que vous devrez peut-être autoriser dans le fichier de mappage.

Mappage des données d'identification pour un serveur de fichiers 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 de protocole, vous pouvez mapper les données d'identification de l'utilisateur dans Managed File Transfer aux données d'identification de l'utilisateur sur le serveur de fichiers 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. Si vous configurez des exits utilisateur de mappage de données d'identification, ils prennent la place de la fonction de mappage de données d'identification par défaut.

Configuration des exits utilisateur de données d'identification de pont de protocole

Pourquoi et quand exécuter cette tâche

Un exit utilisateur pour le mappage des données d'identification de pont de protocole doit implémenter l'une des interfaces suivantes:

- `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit`, qui permet à un agent de pont de protocole de transférer des fichiers vers et depuis un serveur de fichiers de protocole par défaut

- `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit2`, qui vous permet de transférer des fichiers vers et depuis plusieurs noeuds finaux.

L'interface `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit2` contient la même fonction que `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit` et inclut également une fonction étendue. Pour plus d'informations, voir «[Interface ProtocolBridgeCredentialExit.java](#)», à la page 932 et «[Interface ProtocolBridgeCredentialExit2.java](#)», à la page 933.

Les exits de données d'identification peuvent être chaînés de la même manière que les autres exits utilisateur. Les exits sont appelés dans l'ordre dans lequel ils sont spécifiés à l'aide de la propriété `protocolBridgeCredentialConfiguration` dans le fichier de propriétés de l'agent. Les méthodes d'initialisation sont toutes renvoyées séparément et si une ou plusieurs d'entre elles renvoient la valeur false, l'agent ne démarre pas. L'erreur est signalée dans le journal des événements de l'agent.

Un seul résultat global est renvoyé pour les méthodes d'ID `mapMQUser` de tous les exits comme suit:

- Si la méthode renvoie la valeur `USER_SUCCESSFULLY_MAPPING` ou `USER_NIGED_ACCESS` comme code de résultat, cette valeur est le résultat renvoyé et les méthodes d'ID `mapMQUser` des exits suivants ne sont pas appelées.
- Si la méthode renvoie la valeur `NO_MAPPING_FOUND` comme code de résultat, la méthode d'ID `mqMQUser` de l'exit suivant est appelée.
- S'il n'y a pas d'exit suivant, le résultat `NO_MAPPING_FOUND` est renvoyé.
- Un code de résultat global de `USER_DÉNIÉD_ACCESS` ou `NO_MAPPING_FOUND` est considéré comme un échec de transfert par l'agent de pont.

Pour exécuter votre exit, procédez comme suit:

Procédure

1. Compilez l'exit utilisateur des données d'identification du pont de protocole.
2. Créez un fichier d'archive Java (JAR) contenant l'exit compilé et sa structure de package.
3. Placez le fichier JAR qui contient la classe d'exit dans le répertoire `exits` de l'agent de pont. Le répertoire se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name`.
4. Editez le fichier de propriétés de l'agent de pont de protocole pour inclure la propriété `protocolBridgeCredentialExitClasses`. Pour la valeur de cette propriété, spécifiez une liste de classes séparées par des virgules qui implémentent une routine d'exit de données d'identification de pont de protocole. Les classes d'exit sont appelées dans l'ordre dans lequel elles sont spécifiées dans cette liste. Pour plus d'informations, voir «[Le fichier MFT agent.properties](#)», à la page 564.
5. Editez le fichier de propriétés de l'agent de pont de protocole pour inclure:

```
exitClassPath=IBM MQ
installation_directory\mqft\config\configuration_queue_manager\agents\protocol_bridge_agent_n
ame\exits\SampleCredentialExit.jar
```

Le fichier `agent.properties` d'un agent se trouve dans votre répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/bridge_agent_name`.

Si vous modifiez le fichier `agent.properties`, vous devez redémarrer l'agent pour appliquer les modifications.

6. Vous pouvez éventuellement spécifier la propriété `protocolBridgeCredentialConfiguration`. La valeur que vous spécifiez pour cette propriété est transmise en tant qu'objet `String` à la méthode `initialize()` des classes d'exit spécifiées par `protocolBridgeCredentialExitClasses`. Pour plus d'informations, voir «[Le fichier MFT agent.properties](#)», à la page 564.
7. Démarrez l'agent de pont de protocole à l'aide de la commande **`fteStartAgent`**.

Pourquoi et quand exécuter cette tâche

Un exemple d'exit de données d'identification de pont de protocole est fourni dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/protocolBridge` et dans la rubrique «Exemple d'exit utilisateur de données d'identification de pont de protocole», à la page 271. Cet exemple est basé sur l'interface `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit`.

L'exit `SampleCredentialExit.java` lit un fichier de propriétés qui mappe les ID utilisateur MQMD associés aux demandes de transfert vers les ID utilisateur du serveur et les mots de passe du serveur. L'emplacement du fichier de propriétés est extrait de la propriété d'agent de pont de protocole `protocolBridgeCredentialConfiguration`.

Pour exécuter l'exemple d'exit utilisateur, procédez comme suit:

Procédure

1. Compilez le fichier `SampleCredentialExit.java`.
2. Créez un fichier JAR contenant l'exit compilé et sa structure de package.
3. Placez le fichier JAR dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name/exits`.
4. Editez le fichier `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name/agent.properties` pour qu'il contienne la ligne suivante:

```
protocolBridgeCredentialExitClasses=SampleCredentialExit
```

5. Editez le fichier de propriétés de l'agent de pont de protocole pour inclure:

```
exitClassPath=IBM MQ
installation_directory\mqft\config\configuration_queue_manager\agents\protocol_bridge_agent_n
ame\exits\SampleCredentialExit.jar
```

Le fichier `agent.properties` d'un agent se trouve dans votre répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name`.

Si vous modifiez le fichier `agent.properties`, vous devez redémarrer l'agent pour appliquer les modifications.

6. Créez un fichier de propriétés de données d'identification (`credentials.properties`) dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent` et éditez-le pour inclure des entrées au format suivant:

```
mqUserId=serverUserId,serverPassword
```

7. Editez le fichier `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name/agent.properties` pour qu'il contienne la ligne suivante:

```
protocolBridgeCredentialConfiguration=MQ_DATA_PATH/mqft/
config/coordination_queue_manager/agents/bridge_agent_name/credentials.properties
```

Vous devez utiliser le chemin d'accès absolu au fichier `credentials.properties`.

8. Démarrez l'agent de pont de protocole à l'aide de la commande **fteStartAgent**.

Exemple: Comment configurer un agent de pont de protocole pour utiliser des données d'identification de clé privée avec un serveur SFTP UNIX

Cet exemple explique comment générer et configurer le fichier `ProtocolBridgeCredentials.xml`. Cet exemple est typique et les détails peuvent varier en fonction de votre plateforme, mais les principes restent les mêmes.

Pourquoi et quand exécuter cette tâche

Procédure

1. Générez une clé publique et une clé privée à utiliser pour l'authentification auprès du serveur SFTP.

Par exemple, sur un système hôte Linux, vous pouvez utiliser l'outil **ssh-keygen**, fourni avec le package 'openssh', pour créer la paire clé publique / clé privée.

Par défaut, sans arguments, la commande **ssh-keygen** vous invite à indiquer un emplacement et une phrase passe pour les deux fichiers de clés, dont les noms par défaut sont les suivants:

```
id_rsa      <-- Private key
id_rsa.pub  <-- Public key
```



Avertissement : Si vous utilisez la commande **ssh-keygen** à partir d'une version récente d'OpenSSH, telle que celle fournie avec RHEL 8, le format de clé utilisé n'est pas compatible avec l'agent de pont de protocole et les tentatives de transfert vers le serveur SFTP échouent avec le message suivant:

```
BFGBR0216E: Authentication to protocol server 'sftp.host.address' failed
because of invalid private key.
```

Pour créer une clé privée compatible avec ces nouvelles versions d'OpenSSH, spécifiez le format de clé avec l'argument suivant dans la commande **ssh-keygen** :

```
ssh-keygen -m PEM
```

Le contenu de la clé privée `id_rsa` contient alors les première et dernière lignes de:

```
-----BEGIN RSA PRIVATE KEY-----
...
-----END RSA PRIVATE KEY-----
```

qui est compatible avec l'agent de pont de protocole.

2. Copiez l'intégralité du contenu du fichier `id_rsa.pub` dans le fichier `~/.ssh/authorized_keys` de l'utilisateur SFTP sur le serveur SFTP.

Vérifiez que les droits d'accès à ce fichier et au répertoire `~/.ssh` sont définis de manière appropriée pour que le serveur SFTP autorise l'authentification par clé. Ces droits sont généralement les suivants:

```
~/.ssh          Mode 700
~/.ssh/authorized_keys  Mode 600
```

3. Managed File Transfer requiert une empreinte digitale SSH d'hôte générée à l'aide de l'algorithme MD5. Exécutez l'une des commandes suivantes pour obtenir l'empreinte SSL de l'hôte du serveur SFTP.

- Pour Red Hat® Enterprise Linux version 6.x et les versions suivantes, et Linux Ubuntu 14.04, exécutez la commande suivante:

```
ssh-keygen -l -f /etc/ssh/ssh_host_rsa_key.pub
```

- A partir de Red Hat Enterprise Linux version 7.x, Linux Ubuntu 16.04 et SuSE Linux 12.4, la commande `ssh-keygen` génère par défaut l'empreinte digitale ssh à l'aide de l'algorithme SHA56 . Pour générer l'empreinte digitale ssh à l'aide de l'algorithme MD5 , exécutez la commande suivante:

```
ssh-keygen -l -E MD5 -f /etc/ssh/ssh_host_rsa_key.pub
```

La sortie de la commande est similaire à l'exemple suivant:

```
2048 MD5:64:39:f5:49:41:10:55:d2:0b:81:42:5c:87:62:9d:27 no comment (RSA)
```

Extrayez la partie hexadécimale uniquement de la sortie à utiliser comme `hostKey` dans le fichier `ProtocolBridgeCredentials.xml` (voir l'étape «4», à la page 234). Par conséquent, dans cet exemple, vous extrayez `64:39:f5:49:41:10:55:d2:0b:81:42:5c:87:62:9d:27`.

4. Sur le système d'agent de pont de protocole, éditez le fichier `ProtocolBridgeCredentials.xml` . Remplacez les valeurs affichées en italique dans l'exemple suivant par vos propres valeurs:

```
<tns:credentials xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeCredentials"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeCredentials
ProtocolBridgeCredentials.xsd ">

<tns:agent name="Agent_name">

<tns:server name="SFTP_name">

<tns:user name="mq_User_ID" serverUserId="SFTP_user_ID"
hostKey="ssh_host_finger">
<tns:privateKey associationName="name" keyPassword="pass_phrase">
Complete contents of the id_rsa file including the entries
-----BEGIN RSA PRIVATE KEY-----

-----END RSA PRIVATE KEY-----
</tns:privateKey>
</tns:user>

</tns:server>
</tns:agent>
</tns:credentials>
```

où :

- *nom_agent* est le nom de l'agent de pont de protocole.
- *SFTP_host_name* est le nom du serveur SFTP, comme indiqué dans le fichier `ProtocolBridgeProperties.xml` .
- *mq_User_ID* est l'ID utilisateur MQMD associé à la demande de transfert.
- *SFTP_user_ID* est l'ID utilisateur SFTP tel qu'il est utilisé à l'étape 2. Il s'agit de la valeur transmise au SFTP qui sert d'ID utilisateur de connexion.
- *ssh_host_finger* est l'empreinte digitale collectée à l'étape 3.
- *name* est un nom que vous pouvez spécifier pour être utilisé à des fins de trace et de consignation.
- *pass_phrase* est la phrase passe que vous avez fournie dans `ssh-keygen` à l'étape 1.
- *Contenu complet du fichier id_rsa* est le contenu complet du fichier `id_rsa` généré à l'étape 1. Pour éviter une erreur de connexion, veillez à inclure les deux entrées suivantes:

```
-----BEGIN RSA PRIVATE KEY-----
-----END RSA PRIVATE KEY-----
```

Vous pouvez ajouter des clés supplémentaires en dupliquant l'élément `<tns:privatekey>`.

5. Démarrez l'agent de pont de protocole si l'agent n'est pas déjà démarré. Sinon, l'agent de pont de protocole interroge régulièrement le fichier `ProtocolBridgeCredentials.xml` et récupère les modifications.

Configuration d'un pont de protocole pour un serveur FTPS

Configurez un serveur FTPS de la même manière que vous configurez un serveur FTP: créez un agent de pont pour le serveur, définissez les propriétés du serveur et mappez les données d'identification de l'utilisateur.

Pourquoi et quand exécuter cette tâche

Pour configurer un serveur FTPS, procédez comme suit:

Procédure

1. Créez un agent de pont de protocole pour le serveur FTPS à l'aide de la commande **fteCreateBridgeAgent**. Les paramètres applicables à FTP sont également applicables à FTPS, mais il existe également trois paramètres obligatoires spécifiques à FTPS:
 - a) Le paramètre **-bt**. Spécifiez FTPS comme valeur de ce paramètre.
 - b) Paramètre **-bts** du fichier de clés certifiées. La commande suppose que seule l'authentification du serveur est requise et que vous devez spécifier l'emplacement du fichier de clés certifiées.

La forme explicite du protocole FTPS est configurée par la commande **fteCreateBridgeAgent** par défaut, mais vous pouvez configurer la forme implicite en modifiant le fichier de propriétés du pont de protocole. Le pont de protocole se connecte toujours aux serveurs FTPS en mode passif.

Pour plus d'informations sur la commande **fteCreateBridgeAgent**, voir [«fteCreateBridgeAgent \(créer et configurer un agent de pont de protocole MFT\)»](#), à la page 403.

Si vous avez besoin d'instructions pour créer des fichiers de clés certifiées, consultez l'article IBM Developer, [Configuring Secure Sockets Layer connectivity in IBM WebSphere MQ File Transfer Edition](#), ou consultez les informations sur l'outil de clés dans la [documentation Oracle keytool](#).

2. Définissez les propriétés du serveur FTPS dans un élément `<ftpsServer>` dans le fichier de propriétés du pont de protocole: `ProtocolBridgeProperties.xml`. Pour plus d'informations, voir la section [«Définition des propriétés des serveurs de fichiers de protocole à l'aide du fichier ProtocolBridgeProperties.xml»](#), à la page 221. Vous pouvez également activer l'authentification client en éditant le fichier de propriétés du pont de protocole. Pour plus de détails sur toutes les options de configuration, voir [«Format du fichier de propriétés du pont de protocole»](#), à la page 592.
3. Mappez les données d'identification de l'utilisateur dans Managed File Transfer aux données d'identification de l'utilisateur sur le serveur FTPS à l'aide de la fonction de mappage des données d'identification par défaut de l'agent de pont de protocole ou en écrivant votre propre exit utilisateur. Pour plus d'informations, voir la section [«Mappage des données d'identification pour un serveur de fichiers»](#), à la page 228.
4. Par défaut, le fichier de clés certifiées est configuré au format JKS ; si vous souhaitez modifier le format, éditez le fichier de propriétés du pont de protocole.

Exemple

Voici un exemple d'entrée pour un serveur FTPS dans le fichier de propriétés du pont de protocole:

```
<tns:serverProperties xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeProperties"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeProperties
  ProtocolBridgeProperties.xsd">
  <tns:defaultServer name="ftpserver.mycompany.com"/>

  <tns:ftpsServer name="ftpserver.mycompany.com" host="ftpserver.mycompany.com" port="990"
  platform="windows"
  timeZone="Europe/London" locale="en_US" fileEncoding="UTF8"
  listFormat="unix" limitedWrite="false"
  trustStore="c:\mydirec\truststore.jks"/>

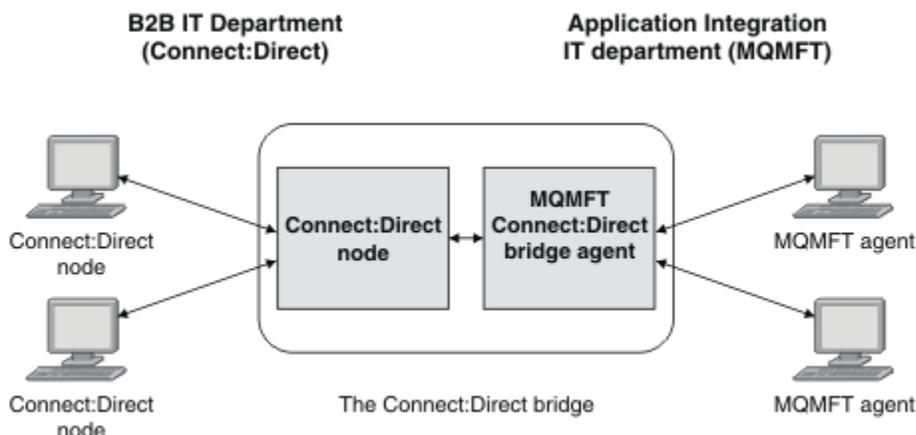
  <!-- Define servers here -->
</tns:serverProperties>
```

Que faire ensuite

Pour plus d'informations sur les parties du protocole FTPS prises en charge et non prises en charge, voir [«Prise en charge du serveur FTPS par le pont de protocole»](#), à la page 685.

Le pont Connect:Direct

Vous pouvez transférer des fichiers depuis et vers un réseau IBM Sterling Connect:Direct existant. Utilisez le pont Connect:Direct, qui est un composant de Managed File Transfer, pour transférer des fichiers entre MFT et IBM Sterling Connect:Direct.



Le diagramme montre un pont MFT Connect:Direct entre deux services, le service informatique B2B et le service informatique Application Integration. Le service informatique B2B utilise Connect:Direct pour transférer des fichiers depuis et vers les partenaires commerciaux de la société. Le service informatique d'intégration d'applications utilise IBM MQ comme infrastructure de messagerie et par conséquent, a récemment choisi Managed File Transfer comme solution de transfert de fichier.

A l'aide du pont MFT Connect:Direct, les deux services peuvent transférer des fichiers entre le réseau Connect:Direct du service informatique B2B et le réseau MFT du service informatique Application Integration. Le pont Connect:Direct est un composant de Managed File Transfer, qui inclut un agent MFT qui communique avec un noeud Connect:Direct. L'agent MFT est dédié aux transferts avec le noeud Connect:Direct : il s'agit de l'agent de pont Connect:Direct.

Le pont Connect:Direct est disponible dans les composants Service et Agent de Managed File Transfer et peut être utilisé pour les tâches suivantes :

1. Utiliser des commandes Managed File Transfer pour initier le transfert d'un fichier ou de plusieurs fichiers depuis un agent MFT vers un noeud Connect:Direct.
2. Utiliser des commandes Managed File Transfer pour initier le transfert d'un fichier ou de plusieurs fichiers depuis un noeud Connect:Direct vers un agent MFT.
3. Utiliser des commandes Managed File Transfer pour initier le transfert d'un fichier qui lance un processus Connect:Direct défini par l'utilisateur.
4. Utiliser un processus Connect:Direct pour soumettre une demande de transfert de fichier MFT.

Un pont Connect:Direct peut transférer des fichiers vers ou depuis des noeuds Connect:Direct seulement. Le pont Connect:Direct peut transférer des fichiers vers ou depuis son système de fichiers local uniquement dans le cadre d'un transfert soumis par un processus Connect:Direct .

z/OS Vous pouvez utiliser le pont Connect:Direct pour effectuer un transfert vers ou depuis un jeu de données qui se trouve sur un noeud Connect:Direct sur un système z/OS. Le comportement est différent de celui des transferts de jeu de données qui n'impliquent que des agents Managed File Transfer. Pour plus d'informations, voir [«Transfert de fichiers vers et depuis des noeuds Connect:Direct»](#), à la page 653.

Plateformes prises en charge

Le pont Connect:Direct est constitué d'un agent de pont MFT Connect:Direct et d'un noeud Connect:Direct . L'agent est pris en charge sous Windows et Linux for System x. Le noeud est pris en charge sur les plateformes qui sont prises en charge pour IBM Sterling Connect:Direct for Windows et IBM Sterling Connect:Direct for UNIX. Pour des instructions de création de l'agent de pont Connect:Direct et de configuration d'un noeud Connect:Direct pour que l'agent puisse communiquer avec lui, voir «[Configuration du pont Connect:Direct](#)», à la page 146.

Le pont Connect:Direct peut transférer des fichiers depuis et vers des noeuds Connect:Direct qui s'exécutent dans le cadre de l'installation de service Connect:Direct for Windows, Connect:Direct for UNIX  ou Connect:Direct for z/OS. Pour des détails sur les versions de Connect:Direct qui sont prises en charge, voir la page Web [IBM MQ System Requirements](#).

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 System x
- L'agent se trouve sous Linux for System x et le noeud sous UNIX
- 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 System x 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.

Transfert d'un fichier vers un noeud Connect:Direct

Vous pouvez transférer un fichier d'un agent Managed File Transfer vers un noeud Connect:Direct à l'aide du pont Connect:Direct . Indiquez un noeud Connect:Direct comme destination du transfert en spécifiant l'agent de pont Connect:Direct comme agent de destination et en spécifiant le fichier de destination sous la forme `connect_direct_node_name:file_path`.

Avant de commencer

Avant de transférer un fichier, vous devez configurer le pont Connect:Direct , qui est un composant de Managed File Transfer. Pour plus d'informations, voir la section «[Configuration du pont Connect:Direct](#)», à la page 146.

Pourquoi et quand exécuter cette tâche

Dans cet exemple, l'agent de pont Connect:Direct est appelé CD_BRIDGE. L'agent source est appelé FTE_AGENT et peut être n'importe quelle version de WMQFTE. Le noeud Connect:Direct de destination est appelé CD_NODE1. Le fichier à transférer se trouve dans le chemin d'accès au fichier `/home/helen/file.log` sur le système où se trouve FTE_AGENT. Le fichier est transféré dans le chemin d'accès au fichier `/files/data.log` sur le système sur lequel CD_NODE1 s'exécute.

Procédure

1. Utilisez la commande de transfert `fteCreate` avec la valeur du paramètre **-df** (fichier de destination) sous la forme `connect_direct_node_name:file_path` et la valeur du paramètre **-da** (agent de destination) spécifiée comme nom de l'agent de pont Connect:Direct .

Remarque : Le noeud Connect:Direct spécifié par *connect_direct_node_name* est le noeud vers lequel le fichier doit être transféré, et non le noeud Connect:Direct qui fait partie du pont Connect:Direct .

```
fteCreateTransfer -sa FTE_AGENT -da CD_BRIDGE  
-df CD_NODE1:/files/data.log /home/helen/file.log
```

Pour plus d'informations, voir la section [«fteCreateTransfer: démarrer un nouveau transfert de fichier»](#), à la page 447.

2. L'agent source FTE_AGENT transfère le fichier à l'agent de pont Connect:Direct CD_BRIDGE. Le fichier est temporairement stocké sur le système où s'exécute l'agent de pont Connect:Direct , à l'emplacement défini par la propriété d'agent cdTmpDir. L'agent de pont Connect:Direct transfère le fichier au noeud Connect:Direct CD_NODE1.

Concepts associés

[«Le pont Connect:Direct»](#), à la page 236

Vous pouvez transférer des fichiers depuis et vers un réseau IBM Sterling Connect:Direct existant. Utilisez le pont Connect:Direct, qui est un composant de Managed File Transfer, pour transférer des fichiers entre MFT et IBM Sterling Connect:Direct.

Tâches associées

[«Transfert d'un fichier à partir d'un noeud Connect:Direct»](#), à la page 238

Vous pouvez transférer un fichier d'un noeud Connect:Direct vers un Managed File Transfer Agent à l'aide du pont Connect:Direct . Vous pouvez spécifier un noeud Connect:Direct comme source du transfert en spécifiant l'agent de pont Connect:Direct comme agent source et en spécifiant la spécification source sous la forme *connect_direct_node_name:file_path*.

Référence associée

[«Le fichier MFT agent.properties»](#), à la page 564

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.

Transfert d'un fichier à partir d'un noeud Connect:Direct

Vous pouvez transférer un fichier d'un noeud Connect:Direct vers un Managed File Transfer Agent à l'aide du pont Connect:Direct . Vous pouvez spécifier un noeud Connect:Direct comme source du transfert en spécifiant l'agent de pont Connect:Direct comme agent source et en spécifiant la spécification source sous la forme *connect_direct_node_name:file_path*.

Avant de commencer

Avant de transférer un fichier, vous devez configurer le pont Connect:Direct , qui est un composant de Managed File Transfer. Voir [«Configuration du pont Connect:Direct»](#), à la page 146.

Pourquoi et quand exécuter cette tâche

Dans cet exemple, l'agent de pont Connect:Direct est appelé CD_BRIDGE. L'agent de destination est appelé FTE_AGENT et peut être n'importe quelle version de Managed File Transfer. Le noeud Connect:Direct source est appelé CD_NODE1. Le fichier à transférer se trouve dans le chemin d'accès au fichier */home/brian/in.file* sur le système où se trouve CD_NODE1 . Le fichier est transféré dans le chemin d'accès au fichier */files/out.file* sur le système où FTE_AGENT est en cours d'exécution.

Procédure

Utilisez la commande **fteCreateTransfer** avec la valeur de la spécification source sous la forme *connect_direct_node_name:file_path* et la valeur du paramètre **-sa** spécifiée comme nom de l'agent de pont Connect:Direct .

Remarque : Le noeud Connect:Direct spécifié par *connect_direct_node_name* est le noeud à partir duquel le fichier doit être transféré, et non le noeud Connect:Direct qui fait partie du pont Connect:Direct. Exemple :

```
fteCreateTransfer -sa CD_BRIDGE -da FTE_AGENT  
-df /files/out.file CD_NODE1:/home/brian/in.file
```

Pour plus d'informations, voir la section [«fteCreateTransfer: démarrer un nouveau transfert de fichier»](#), à la page 447.

Résultats

L'agent de pont Connect:Direct CD_BRIDGE demande le fichier au noeud Connect:Direct CD_NODE1. Le noeud Connect:Direct envoie le fichier au pont Connect:Direct. Lors du transfert du fichier à partir du noeud Connect:Direct, le pont Connect:Direct stocke temporairement le fichier à l'emplacement défini par la propriété d'agent *cdTmpDir*. Lorsque le transfert du fichier depuis le noeud Connect:Direct vers le pont Connect:Direct est terminé, le pont Connect:Direct envoie ensuite le fichier à l'agent cible FTE_AGENT et le supprime de l'emplacement temporaire.

Concepts associés

«Le pont Connect:Direct», à la page 236

Vous pouvez transférer des fichiers depuis et vers un réseau IBM Sterling Connect:Direct existant. Utilisez le pont Connect:Direct, qui est un composant de Managed File Transfer, pour transférer des fichiers entre MFT et IBM Sterling Connect:Direct.

Référence associée

«Le fichier MFT *agent.properties*», à la page 564

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.

Transfert d'un ensemble de données vers un noeud Connect:Direct sur z/OS

Vous pouvez transférer un ensemble de données d'un agent Managed File Transfer sous z/OS vers un noeud Connect:Direct sous z/OS à l'aide d'un pont Connect:Direct situé sur un système Windows ou Linux.

Avant de commencer

Avant de transférer un fichier, vous devez configurer le pont Connect:Direct, qui est un composant de Managed File Transfer. Voir [«Configuration du pont Connect:Direct»](#), à la page 146.

Pourquoi et quand exécuter cette tâche

Dans cet exemple, le paramètre **-df** est utilisé pour spécifier la destination du transfert. Le paramètre **-df** peut être utilisé lorsque l'agent source du transfert est une version quelconque de Managed File Transfer. Si l'agent source est IBM WebSphere MQ File Transfer Edition 7.0.4 ou une version ultérieure, vous pouvez utiliser le paramètre **-ds** à la place. L'agent source est appelé FTE_ZOS1 et est un agent IBM WebSphere MQ File Transfer Edition 7.0.3. L'agent de pont Connect:Direct est appelé CD_BRIDGE et se trouve sur un système Linux. Le noeud Connect:Direct de destination est appelé CD_ZOS2. L'agent source et le noeud Connect:Direct de destination se trouvent sur les systèmes z/OS. Le fichier à transférer se trouve dans *//FTEUSER.SOURCE.LIB* sur le système où se trouve FTE_ZOS1. Le fichier est transféré au fichier *//CDUSER.DEST.LIB* sur le système où se trouve CD_ZOS2.

Procédure

1. Utilisez la commande de transfert *fteCreate* avec la valeur du paramètre **-df** sous la forme *connect_direct_node_name:data_set_name;attributes* et la valeur du paramètre **-da** (agent de destination) spécifiée comme nom de l'agent de pont Connect:Direct.

Le noeud Connect:Direct spécifié par *connect_direct_node_name* est le noeud vers lequel vous souhaitez transférer l'ensemble de données, et non le noeud Connect:Direct qui fait partie du pont Connect:Direct .

Le nom de fichier spécifié par *nom_fichier* doit être absolu et non relatif. Connect:Direct ne préfixe pas le nom de l'ensemble de données avec le nom de l'utilisateur.

```
fteCreateTransfer -sa FTE_ZOS1 -sm QM_ZOS
                 -da CD_BRIDGE -dm QM_BRIDGE
                 -df CD_ZOS2: //'CDUSER.DEST.LIB;BLKSIZE(8000);LRECL(80)'
                 //'FTEUSER.SOURCE.LIB'
```

Pour plus d'informations, voir la section [«fteCreateTransfer: démarrer un nouveau transfert de fichier»](#), à la page 447.

2. L'agent source FTE_ZOS1 transfère les données du jeu de données à l'agent de pont Connect:Direct CD_BRIDGE. Les données sont temporairement stockées sous forme de fichier à plat sur le système où s'exécute l'agent de pont Connect:Direct , à l'emplacement défini par la propriété d'agent cdTmpDir. L'agent de pont Connect:Direct transfère les données au noeud Connect:Direct CD_ZOS2. Une fois le transfert terminé, le fichier à plat est supprimé du système sur lequel l'agent de pont Connect:Direct est en cours d'exécution.

Concepts associés

[«Le pont Connect:Direct»](#), à la page 236

Vous pouvez transférer des fichiers depuis et vers un réseau IBM Sterling Connect:Direct existant. Utilisez le pont Connect:Direct, qui est un composant de Managed File Transfer, pour transférer des fichiers entre MFT et IBM Sterling Connect:Direct.

Référence associée

[«Transfert de fichiers vers et depuis des noeuds Connect:Direct»](#), à la page 653

Vous pouvez transférer des ensembles de données entre les agents Managed File Transfer et les noeuds IBM Sterling Connect:Direct à l'aide du pont Connect:Direct . Vous pouvez spécifier un fichier comme source de transfert, destination de transfert ou les deux.

[«Mappages entre les paramètres d'instruction de processus Connect:Direct et les clés BPXWDYN»](#), à la page 655

Lorsque vous soumettez une demande de transfert pour un fichier dont la source ou la destination est un noeud Connect:Direct , toutes les clés BPXWDYN prises en charge que vous fournissez sont converties dans un format accepté par les processus Connect:Direct .

Transfert de plusieurs fichiers vers un noeud Connect:Direct

Vous pouvez transférer plusieurs fichiers d'un Managed File Transfer Agent vers un noeud Connect:Direct à l'aide du pont Connect:Direct . Pour utiliser un noeud Connect:Direct comme destination du transfert de plusieurs fichiers, spécifiez l'agent de pont Connect:Direct comme agent de destination et indiquez le répertoire de destination sous la forme *connect_direct_node_name:directory_path*.

Avant de commencer

Avant de transférer des fichiers, vous devez configurer le pont Connect:Direct , qui est un composant de Managed File Transfer. Voir [«Configuration du pont Connect:Direct»](#), à la page 146.

Pourquoi et quand exécuter cette tâche

Dans cet exemple, l'agent source est appelé FTE_AGENT. L'agent de pont Connect:Direct est appelé CD_BRIDGE. Le noeud Connect:Direct de destination est appelé CD_NODE1. Les fichiers à transférer sont /home/jack/data.log, /logs/log1.txt et /results/latest sur le système où se trouve FTE_AGENT. Les fichiers sont transférés vers le répertoire /in/files sur le système sur lequel CD_NODE1 s'exécute.

Procédure

Utilisez la commande de transfert `fteCreate` avec la valeur du paramètre **-dd** (répertoire de destination) sous la forme `connect_direct_node_name:directory_path`. Indiquez la valeur du paramètre **-da** (agent de destination) comme nom de l'agent de pont `Connect:Direct`.

Remarque : Le noeud `Connect:Direct` spécifié par `connect_direct_node_name` est le noeud vers lequel vous souhaitez que les fichiers soient transférés, et non le noeud `Connect:Direct` qui fait partie du pont `Connect:Direct`.

```
fteCreateTransfer -sa FTE_AGENT -da CD_BRIDGE
                 -dd CD_NODE1:/in/files /home/jack/data.log
                 /logs/log1.txt /results/latest
```

Pour plus d'informations, voir la section [«fteCreateTransfer: démarrer un nouveau transfert de fichier»](#), à la page 447.

Résultats

L'agent source `FTE_AGENT` transfère le premier fichier à l'agent de pont `Connect:Direct CD_BRIDGE`. L'agent de pont `Connect:Direct` stocke temporairement le fichier à l'emplacement défini par la propriété de répertoire `cdTmp`. Lorsque le fichier a été intégralement transféré de l'agent source vers le pont `Connect:Direct`, l'agent de pont `Connect:Direct` envoie le fichier au noeud `Connect:Direct` défini par la propriété d'agent `cdNode`. Ce noeud envoie le fichier au noeud `Connect:Direct` de destination `CD_NODE1`. L'agent de pont `Connect:Direct` supprime le fichier de l'emplacement temporaire lorsque le transfert entre les deux noeuds `Connect:Direct` est terminé. Ce processus est répété pour chaque fichier source spécifié.

Concepts associés

[«Le pont Connect:Direct»](#), à la page 236

Vous pouvez transférer des fichiers depuis et vers un réseau IBM Sterling `Connect:Direct` existant. Utilisez le pont `Connect:Direct`, qui est un composant de Managed File Transfer, pour transférer des fichiers entre MFT et IBM Sterling `Connect:Direct`.

Tâches associées

[«Transfert d'un fichier vers un noeud Connect:Direct»](#), à la page 237

Vous pouvez transférer un fichier d'un agent Managed File Transfer vers un noeud `Connect:Direct` à l'aide du pont `Connect:Direct`. Indiquez un noeud `Connect:Direct` comme destination du transfert en spécifiant l'agent de pont `Connect:Direct` comme agent de destination et en spécifiant le fichier de destination sous la forme `connect_direct_node_name:file_path`.

[«Transfert de plusieurs fichiers vers Connect:Direct à l'aide de caractères génériques»](#), à la page 243

Pour transférer plusieurs fichiers d'un agent Managed File Transfer vers un noeud `Connect:Direct`, utilisez le pont `Connect:Direct`. Vous pouvez utiliser des caractères génériques dans la spécification source que vous fournissez à la commande **fteCreateTransfer**. Comme pour tous les transferts Managed File Transfer impliquant des caractères génériques, seule la dernière partie du chemin de fichier peut contenir un caractère générique. Par exemple, `/abc/def*` est un chemin de fichier valide et `/abc*/def` n'est pas valide.

[«Transfert d'un fichier à partir d'un noeud Connect:Direct»](#), à la page 238

Vous pouvez transférer un fichier d'un noeud `Connect:Direct` vers un Managed File Transfer Agent à l'aide du pont `Connect:Direct`. Vous pouvez spécifier un noeud `Connect:Direct` comme source du transfert en spécifiant l'agent de pont `Connect:Direct` comme agent source et en spécifiant la spécification source sous la forme `connect_direct_node_name:file_path`.

[«Transfert de plusieurs fichiers à partir d'un noeud Connect:Direct»](#), à la page 242

Vous pouvez transférer plusieurs fichiers d'un noeud `Connect:Direct` vers un Managed File Transfer Agent à l'aide du pont `Connect:Direct`. Vous pouvez spécifier un noeud `Connect:Direct` comme source du transfert de plusieurs fichiers en spécifiant l'agent de pont `Connect:Direct` comme agent source et en spécifiant une ou plusieurs spécifications de source sous la forme `connect_direct_node_name:file_path`.

Référence associée

[«Le fichier MFT agent.properties»](#), à la page 564

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.

Transfert de plusieurs fichiers à partir d'un noeud Connect:Direct

Vous pouvez transférer plusieurs fichiers d'un noeud Connect:Direct vers un Managed File Transfer Agent à l'aide du pont Connect:Direct . Vous pouvez spécifier un noeud Connect:Direct comme source du transfert de plusieurs fichiers en spécifiant l'agent de pont Connect:Direct comme agent source et en spécifiant une ou plusieurs spécifications de source sous la forme `connect_direct_node_name:file_path`.

Avant de commencer

Avant de transférer un fichier, vous devez configurer le pont Connect:Direct , qui est un composant de Managed File Transfer. Voir [«Configuration du pont Connect:Direct»](#), à la page 146.

Pourquoi et quand exécuter cette tâche

Dans cet exemple, l'agent de pont Connect:Direct est appelé `CD_BRIDGE`. L'agent de destination est appelé `FTE_Z` et s'exécute sur un système z/OS . Le noeud Connect:Direct source est appelé `CD_NODE1`. Les fichiers à transférer se trouvent dans les chemins d'accès aux fichiers `/in/file1`, `/in/file2` et `/in/file3` sur le système où se trouve `CD_NODE1` . Les fichiers sont transférés vers le fichier partitionné `//OBJECT.LIB` sur le système sur lequel `FTE_Z` s'exécute.

Procédure

Utilisez la commande de transfert `fteCreate` avec les valeurs des spécifications source au format `connect_direct_node_name:file_path` et la valeur du paramètre `-sa` spécifiée comme nom de l'agent de pont Connect:Direct .

Remarque : Le noeud Connect:Direct spécifié par `connect_direct_node_name` est le noeud à partir duquel vous souhaitez que les fichiers soient transférés, et non le noeud Connect:Direct qui fait partie du pont Connect:Direct .

```
fteCreateTransfer -sa CD_BRIDGE -da FTE_Z
                  -dp '//OBJECT.LIB' CD_NODE1:/in/file1
                  CD_NODE1:/in/file2 CD_NODE1:/in/file3
```

Pour plus d'informations, voir la section [«fteCreateTransfer: démarrer un nouveau transfert de fichier»](#), à la page 447.

Résultats

L'agent de pont Connect:Direct `CD_BRIDGE` demande le premier fichier du noeud Connect:Direct `CD_NODE1`. Le noeud Connect:Direct envoie le fichier au pont Connect:Direct . Lors du transfert du fichier à partir du noeud Connect:Direct , le pont Connect:Direct stocke temporairement le fichier à l'emplacement défini par la propriété d'agent `cdTmpDir` . Lorsque le transfert du fichier depuis le noeud Connect:Direct vers le pont Connect:Direct est terminé, le pont Connect:Direct envoie le fichier à l'agent cible `FTE_Z`, puis supprime le fichier de l'emplacement temporaire. Ce processus est répété pour chaque fichier source spécifié.

Concepts associés

[«Le pont Connect:Direct»](#), à la page 236

Vous pouvez transférer des fichiers depuis et vers un réseau IBM Sterling Connect:Direct existant. Utilisez le pont Connect:Direct, qui est un composant de Managed File Transfer, pour transférer des fichiers entre MFT et IBM Sterling Connect:Direct.

Référence associée

[«Le fichier MFT agent.properties»](#), à la page 564

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.

Transfert de plusieurs fichiers vers Connect:Direct à l'aide de caractères génériques

Pour transférer plusieurs fichiers d'un agent Managed File Transfer vers un noeud Connect:Direct, utilisez le pont Connect:Direct. Vous pouvez utiliser des caractères génériques dans la spécification source que vous fournissez à la commande `fteCreateTransfer`. Comme pour tous les transferts Managed File Transfer impliquant des caractères génériques, seule la dernière partie du chemin de fichier peut contenir un caractère générique. Par exemple, `/abc/def*` est un chemin de fichier valide et `/abc*/def` n'est pas valide.

Avant de commencer

Avant de transférer un fichier, vous devez configurer le pont Connect:Direct, qui est un composant de Managed File Transfer. Pour plus d'informations, voir la section [«Configuration du pont Connect:Direct»](#), à la page 146.

Pourquoi et quand exécuter cette tâche

Dans cet exemple, l'agent source est appelé `FTE_AGENT` et l'agent de pont Connect:Direct est appelé `CD_BRIDGE`. Le noeud Connect:Direct de destination est appelé `CD_NODE1`. Les fichiers à transférer se trouvent dans le répertoire `/reports` sur le système où se trouve `FTE_AGENT`. Seuls les fichiers dont le nom commence par `report`, suivi de deux caractères et du suffixe `.log`, sont transférés. Par exemple, le fichier `/reports/report01.log` est transféré, mais le fichier `/reports/report1.log` n'est pas transféré. Les fichiers sont transférés vers le répertoire `/home/fred` sur le système sur lequel `CD_NODE1` s'exécute.

Procédure

1. Utilisez la commande de transfert `fteCreateavec` la valeur du paramètre `-dd` (répertoire de destination) sous la forme `connect_direct_node_name:directory_path`. Pour le paramètre `-da` (agent de destination), spécifiez l'agent de pont Connect:Direct.

Remarque : Le noeud Connect:Direct spécifié par `connect_direct_node_name` est le noeud vers lequel vous souhaitez que les fichiers soient transférés, et non le noeud Connect:Direct qui fait partie du pont Connect:Direct.

```
fteCreateTransfer -sa FTE_AGENT -da CD_BRIDGE
-dd CD_NODE1:/home/fred "/reports/report??.log"
```

Pour plus d'informations, voir la section [«fteCreateTransfer: démarrer un nouveau transfert de fichier»](#), à la page 447.

2. L'agent source `FTE_AGENT` transfère le premier fichier qui correspond au modèle `/reports/report??.log` à l'agent de pont Connect:Direct `CD_BRIDGE`. L'agent de pont Connect:Direct stocke temporairement le fichier à l'emplacement défini par la propriété de répertoire `cdTmp`. Lorsque le fichier a été intégralement transféré de l'agent source vers le pont Connect:Direct, l'agent de pont Connect:Direct envoie le fichier au noeud Connect:Direct défini par la propriété d'agent `cdNode`. Ce noeud envoie le fichier au noeud Connect:Direct de destination `CD_NODE1`. L'agent de pont Connect:Direct supprime le fichier de l'emplacement temporaire lorsque le transfert entre les deux noeuds Connect:Direct est terminé. Ce processus est répété pour chaque fichier source qui correspond au modèle de caractère générique `/reports/report??.log`.

Remarque : La liste des fichiers correspondant au modèle `/reports/report??.log` varie en fonction du système d'exploitation du système sur lequel se trouve l'agent source `FTE_AGENT`.

- Si l'agent source se trouve sur un système avec un système d'exploitation Windows, la correspondance de modèle n'est pas sensible à la casse. Le masque correspond à tous les fichiers

du répertoire `/reports` avec un nom de fichier au format `report` suivi de deux caractères et d'un suffixe `.log`, quelle que soit la casse des lettres. Par exemple, `Report99.Log` est une correspondance.

- Si l'agent source se trouve sur un système Linux ou UNIX, la correspondance de modèle est sensible à la casse. Le modèle correspond uniquement aux fichiers du répertoire `/reports` avec un nom de fichier au format `report` suivi de deux caractères et d'un suffixe `.log`. Par exemple, `reportAB.log` est une correspondance, mais `reportAB.LOG` et `Report99.Log` ne sont pas des correspondances.

Reprise et redémarrage pour les transferts vers et depuis les noeuds Connect:Direct

Il se peut que Managed File Transfer ne puisse pas se connecter à votre noeud IBM Sterling Connect:Direct lors d'un transfert ; par exemple, si le noeud devient indisponible. Soit Managed File Transfer tente de récupérer le transfert, soit le transfert échoue et un message d'erreur est généré.

Si le noeud Connect:Direct devient indisponible

Si le noeud Connect:Direct devient indisponible ; par exemple, en raison d'une panne de réseau ou d'alimentation, Managed File Transfer récupère un transfert de fichiers de l'une des manières suivantes:

- Si Managed File Transfer n'a pas réussi à se connecter au noeud Connect:Direct dans le cadre de cette demande de transfert, le transfert est relancé pendant une période déterminée par les valeurs de **cdMaxConnectionRetries** et **recoverableTransferRetryInterval properties**. Ces propriétés sont spécifiées dans le fichier `agent.properties` pour l'agent de pont Connect:Direct. Le transfert échoue et un message d'erreur est généré lorsque le nombre de tentatives ayant échoué atteint la valeur de **cdMaxConnectionRetries property**. Par défaut, le transfert est tenté indéfiniment, avec 60 secondes entre les tentatives.
- Si Managed File Transfer a précédemment réussi à se connecter au noeud Connect:Direct dans le cadre de cette demande de transfert, le transfert est retenté pendant une période déterminée par les valeurs des propriétés **cdMaxPartialWorkConnectionRetries** et **recoverableTransferRetryInterval**. Le transfert échoue et un message d'erreur est généré lorsque le nombre de tentatives ayant échoué atteint la valeur de la propriété **cdMaxPartialWorkConnectionRetries**. Par défaut, le transfert est tenté indéfiniment, avec 60 secondes entre les tentatives.
- Pour certains types d'échec de noeud Connect:Direct, par exemple le noeud en cours d'arrêt forcé, les processus Connect:Direct passent à l'état `HEld Due to Error (HE)` lors de la reprise du noeud. Une fois le noeud restauré, Managed File Transfer reprend automatiquement tous les processus Connect:Direct liés au transfert de fichiers et dont le statut est HE.
- Si le transfert échoue, tous les fichiers temporaires associés au transfert sont supprimés du système qui héberge le pont Connect:Direct. L'emplacement de ces fichiers temporaires est défini par la propriété **cdTmpDir**.
- Si le transfert est de Managed File Transfer vers Connect:Direct et qu'une disposition source de suppression est spécifiée, les fichiers source ne sont pas supprimés si le transfert échoue.

Si les données d'identification de l'utilisateur du noeud Connect:Direct ne sont pas valides

Si Managed File Transfer ne parvient pas à se connecter au noeud Connect:Direct car les données d'identification de l'utilisateur sont rejetées par le noeud, le transfert échoue et un message d'erreur est généré. Dans ce cas, vérifiez que vous avez fourni les données d'identification utilisateur correctes pour le noeud Connect:Direct. Pour plus d'informations, voir la section [«Mappage des données d'identification pour Connect:Direct»](#), à la page 149.

Si l'agent de pont Connect:Direct devient indisponible

Si l'agent de pont Connect:Direct devient indisponible, les transferts de fichiers en cours sont effectués de la même manière que les transferts Managed File Transfer standard. Pour plus d'informations, voir la section [«MFT Reprise et redémarrage»](#), à la page 252.

Concepts associés

[«Le pont Connect:Direct»](#), à la page 236

Vous pouvez transférer des fichiers depuis et vers un réseau IBM Sterling Connect:Direct existant. Utilisez le pont Connect:Direct, qui est un composant de Managed File Transfer, pour transférer des fichiers entre MFT et IBM Sterling Connect:Direct.

[«MFT Reprise et redémarrage»](#), à la page 252

Si votre agent ou votre gestionnaire de files d'attente n'est pas disponible pour une raison quelconque, par exemple en raison d'une panne d'alimentation ou de réseau, Managed File Transfer effectue la récupération suivante dans les scénarios suivants:

Tâches associées

[«Configuration du pont Connect:Direct»](#), à la page 146

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

[«Le fichier MFT agent.properties»](#), à la page 564

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.

Soumission d'un processus Connect:Direct défini par l'utilisateur à partir d'une demande de transfert de fichier

Vous pouvez soumettre une demande de transfert pour un transfert qui passe par l'agent de pont Connect:Direct qui appelle un processus Connect:Direct défini par l'utilisateur dans le cadre du transfert de fichiers.

Par défaut, lorsque vous soumettez une demande de transfert de fichier pour un transfert qui passe par le pont Connect:Direct, l'agent de pont Connect:Direct génère le processus Connect:Direct qui est utilisé pour transférer le fichier vers ou depuis le noeud Connect:Direct distant.

Toutefois, vous pouvez configurer l'agent de pont Connect:Direct pour qu'il appelle à la place un processus Connect:Direct défini par l'utilisateur à l'aide du fichier `ConnectDirectProcessDefinition.xml`.

Le fichier `ConnectDirectProcessDefinition.xml`

La commande `fteCreateCDAgent` crée le fichier `ConnectDirectProcessDefinitions.xml` dans le répertoire de configuration de l'agent `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent_name`. Avant de pouvoir appeler des processus Connect:Direct définis par l'utilisateur à partir de l'agent de pont Connect:Direct, vous devez configurer des définitions de processus en éditant ce fichier.

Le fichier définit un ou plusieurs ensembles de processus qui incluent l'emplacement d'un ou de plusieurs processus Connect:Direct appelés dans le cadre d'un transfert. Chaque ensemble de processus inclut un certain nombre de conditions. Si le transfert remplit toutes les conditions de l'ensemble de processus, l'ensemble de processus est utilisé pour spécifier les processus Connect:Direct qui sont appelés par le transfert. Pour plus d'informations, voir la section [«Spécification du processus Connect:Direct à démarrer à l'aide du fichier ConnectDirectProcessDefinition.xml»](#), à la page 154.

Variables symboliques intrinsèques

Vous pouvez utiliser les variables symboliques intrinsèques définies par Managed File Transfer pour remplacer des valeurs dans des processus Connect:Direct définis par l'utilisateur. Pour respecter la convention de dénomination Connect:Direct, toutes les variables symboliques intrinsèques utilisées par Managed File Transfer ont le format %FTE suivi de cinq caractères alphanumériques en majuscules.

Lors de la création d'un processus pour transférer des fichiers depuis un noeud Connect:Direct sur le système de pont Connect:Direct, vous devez utiliser la variable intrinsèque %FTETFILE comme valeur pour TO FILE dans le processus Connect:Direct. Lors de la création d'un processus pour transférer des fichiers sur un noeud Connect:Direct depuis le système de pont Connect:Direct, vous devez utiliser la variable intrinsèque %FTEFFILE comme valeur de FROM FILE dans le processus Connect:Direct. Ces variables contiennent les chemins d'accès aux fichiers temporaires que l'agent de pont Connect:Direct utilise pour les transferts vers et depuis le réseau Managed File Transfer.

Pour plus d'informations sur les variables symboliques intrinsèques, voir la documentation du produit Connect:Direct.

Exemples de processus Connect:Direct

Managed File Transfer fournit des exemples de processus Connect:Direct. Ces exemples se trouvent dans le répertoire suivant: `MQ_INSTALLATION_PATH/mqft/samples/ConnectDirectProcessTemplates`.

Spécification du processus Connect:Direct à démarrer à l'aide du fichier `ConnectDirectProcessDefinition.xml`

Indiquez le processus Connect:Direct à démarrer dans le cadre d'un transfert Managed File Transfer. Managed File Transfer fournit un fichier XML que vous pouvez éditer pour spécifier des définitions de processus.

Pourquoi et quand exécuter cette tâche

La commande `fteCreateCDAgent` crée le fichier `ConnectDirectProcessDefinitions.xml` dans le répertoire de configuration de l'agent `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent_name`. Avant de pouvoir appeler des processus Connect:Direct définis par l'utilisateur à partir de l'agent de pont Connect:Direct, vous devez configurer des définitions de processus en éditant ce fichier.

Pour chaque processus à appeler dans le cadre d'un transfert via le pont Connect:Direct, procédez comme suit:

Procédure

1. Définissez le processus Connect:Direct que l'agent de pont Connect:Direct doit appeler dans le cadre du transfert et sauvegardez le modèle de processus dans un fichier.
2. Ouvrez le fichier `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent_name/ConnectDirectProcessDefinitions.xml` dans n'importe quel éditeur de texte.
3. Créez un élément `<processSet>`.
4. Dans l'élément `<processSet>`, créez un élément `<condition>`.
5. Dans l'élément `<condition>`, créez un ou plusieurs éléments qui définissent une condition à laquelle la demande de transfert doit correspondre pour appeler le processus Connect:Direct que vous avez défini à l'étape 1. Ces éléments peuvent être des éléments `<match>` ou des éléments `<defined>`.
 - Utilisez un élément `<match>` pour indiquer que la valeur d'une variable doit correspondre à un modèle. Créez l'élément `<match>` avec les attributs suivants:

- `variable` -nom de la variable dont la valeur est comparée. La variable est un symbole intrinsèque. Pour plus d'informations, voir la section [«Variables de substitution à utiliser avec les processus Connect:Direct définis par l'utilisateur»](#), à la page 680.
- `value` -modèle à comparer à la valeur de la variable spécifiée.
- Facultatif: `pattern` -type de canevas utilisé par la valeur de l'attribut `value` . Ce type de canevas peut être wildcard ou regex. Cet attribut est facultatif et la valeur par défaut est wildcard.
- Utilisez un élément `<defined>` pour indiquer qu'une variable doit avoir une valeur définie. Créez l'élément `<defined>` avec l'attribut suivant:
 - `variable` -nom de la variable pour laquelle une valeur doit être définie. La variable est un symbole intrinsèque. Pour plus d'informations, voir la section [«Variables de substitution à utiliser avec les processus Connect:Direct définis par l'utilisateur»](#), à la page 680.

Les conditions spécifiées dans l'élément `<condition>` sont combinées avec un ET logique. Toutes les conditions doivent être remplies pour que l'agent de pont Connect:Direct puisse appeler le processus spécifié par cet élément `<processSet>` . Si vous ne spécifiez pas d'élément `<condition>` , l'ensemble de processus correspond à tous les transferts.

6. Dans l'élément `<processSet>` , créez un élément `<process>` .

7. Dans l'élément `<process>` , créez un élément `<transfer>` .

L'élément `transfer` spécifie le processus Connect:Direct que l'agent de pont Connect:Direct appelle dans le cadre du transfert. Créez l'élément `<transfer>` avec l'attribut suivant:

- `process` - emplacement du processus Connect:Direct que vous avez défini à l'étape 1. L'emplacement de ce fichier est spécifié avec un chemin d'accès absolu ou relatif au répertoire `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent_name` .

Résultats

Lors de la recherche d'une correspondance de condition, l'agent de pont Connect:Direct effectue une recherche 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 146

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 des définitions de processus Connect:Direct»](#), à la page 605

Le fichier `ConnectDirectProcessDefinitions.xml` du répertoire de configuration de l'agent de pont Connect:Direct spécifie le processus Connect:Direct défini par l'utilisateur à démarrer dans le cadre du transfert de fichiers.

[«fteCreateCDAgent \(créer un agent de pont Connect:Direct\)»](#), à la page 413

La commande `fteCreateCDAgent` crée un Managed File Transfer Agent et sa configuration associée à utiliser avec le pont Connect:Direct .

Utilisation de variables symboliques intrinsèques dans les processus Connect:Direct appelés par Managed File Transfer

Vous pouvez appeler un processus Connect:Direct défini par l'utilisateur à partir d'un transfert Managed File Transfer et transmettre des informations du transfert au processus Connect:Direct en utilisant des variables symboliques intrinsèques dans la définition de processus.

Pourquoi et quand exécuter cette tâche

Cet exemple utilise des variables symboliques intrinsèques pour transmettre des informations d'un transfert Managed File Transfer vers un processus Connect:Direct défini par l'utilisateur. Pour plus d'informations sur les variables symboliques intrinsèques utilisées par Managed File Transfer, voir «Variables de substitution à utiliser avec les processus Connect:Direct définis par l'utilisateur», à la page 680.

Dans cet exemple, le fichier est transféré d'un Managed File Transfer Agent à un noeud de pont Connect:Direct . La première partie du transfert est effectuée par Managed File Transfer. La deuxième partie du transfert est effectuée par un processus Connect:Direct défini par l'utilisateur.

Procédure

1. Créez un processus Connect:Direct qui utilise des variables symboliques intrinsèques.

```
%FTEPNAME PROCESS
  SNODE=%FTESNODE
  PNODEID=(%FTEPUSER,%FTEPPASS)
  SNODEID=(%FTESUSER,%FTESPASS)

COPY001 COPY
  FROM (
    FILE=%FTEFFILE
    DISP=%FTEFDISP
  )
  TO (
    FILE=%FTETFILE
    DISP=%FTETDISP
  )
PEND
```

2. Sauvegardez ce processus dans un fichier texte à l'emplacement suivant: *MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent/Example.cdp*
3. Modifiez le fichier *ConnectDirectProcessDefinition.xml* pour inclure une règle qui appelle le processus Connect:Direct que vous avez créé à l'étape 1.

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:cdprocess xmlns:tns="http://wmqfte.ibm.com/ConnectDirectProcessDefinitions"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/
  ConnectDirectProcessDefinitions ConnectDirectProcessDefinitions.xsd">

  <tns:processSet>
    <tns:condition>
      <tns:match variable="%FTESNODE" value="TOBERMORY" pattern="wildcard"/>
    </tns:condition>
    <tns:process>
      <tns:transfer process="Example.cdp"/>
    </tns:process>
  </tns:processSet>

</tns:cdprocess>
```

Dans cet exemple, si une demande de transfert est soumise à l'agent de pont Connect:Direct dont le noeud Connect:Direct source ou de destination est TOBERMORY, le processus *Example.cdp* Connect:Direct est appelé.

4. Soumettez une demande de transfert de fichier qui répond aux conditions que vous avez définies dans le fichier *ConnectDirectProcessDefinition.xml* à l'étape 3.

Par exemple :

```
fteCreateTransfer -sa ORINOCO -da CD_BRIDGE
                 -sm QM_WIMBLEDON -dm QM_COMMON
                 -de overwrite -df TOBERMORY:/home/bulgaria/destination.txt
                 -sd leave c:\bungo\source.txt
```

Dans cet exemple, le noeud Connect:Direct de destination est TOBERMORY. Ce noeud est le noeud secondaire du transfert et la valeur de %FTESNODE est définie sur TOBERMORY. Cette commande correspond à la condition définie dans le fichier ConnectDirectProcessDefinition.xml .

5. Managed File Transfer transfère le fichier source à un emplacement temporaire sur le même système que l'agent de pont Connect:Direct .
6. L'agent de pont Connect:Direct définit les valeurs des variables symboliques intrinsèques à partir des informations de la demande de transfert et des informations de configuration.

Les variables symboliques intrinsèques sont définies sur les valeurs suivantes:

- %FTEPNAME=*process_name* -Cette valeur est un nom de processus de 8 caractères généré par l'agent de pont Connect:Direct .
- %FTESNODE=TOBERMORY-Cette valeur est définie à partir du paramètre **-df** de la commande **fteCreateTransfer** .
- %FTEPUSER, =*primary_node_user* -Ces informations sont extraites du fichier ConnectDirectCredentials.xml .
- %FTEPPASS=*primary_node_user_password* -Ces informations proviennent du fichier ConnectDirectCredentials.xml .
- %FTESUSER, =*secondary_node_user* -Ces informations sont extraites du fichier ConnectDirectCredentials.xml .
- %FTESPASS=*secondary_node_user_password* -Ces informations sont extraites du fichier ConnectDirectCredentials.xml .
- %FTEFFILE =*emplacement_temporaire* -Cette valeur correspond à l'emplacement temporaire du fichier sur le même système que l'agent de pont Connect:Direct .
- %FTEFDISP=leave-Cette valeur est définie à partir du paramètre **-sd** de la commande **fteCreateTransfer** .
- %FTETFILE=/home/bulgaria/destination.txt -Cette valeur est définie à partir du paramètre **-df** de la commande **fteCreateTransfer** .
- %FTETDISP=overwrite-Cette valeur est définie à partir du paramètre **-de** de la commande **fteCreateTransfer** .

7. Le processus Connect:Direct est démarré sur le noeud de pont Connect:Direct . Connect:Direct transfère le fichier de l'emplacement temporaire sur le système de pont Connect:Direct vers la destination /home/bulgaria/destination.txt sur le système où le noeud Connect:Direct TOBERMORY est en cours d'exécution.

Concepts associés

«Soumission d'un processus Connect:Direct défini par l'utilisateur à partir d'une demande de transfert de fichier», à la page 245

Vous pouvez soumettre une demande de transfert pour un transfert qui passe par l'agent de pont Connect:Direct qui appelle un processus Connect:Direct défini par l'utilisateur dans le cadre du transfert de fichiers.

Référence associée

«Variables de substitution à utiliser avec les processus Connect:Direct définis par l'utilisateur», à la page 680

Vous pouvez définir des valeurs pour remplacer des processus Connect:Direct définis par l'utilisateur à l'aide de variables symboliques intrinsèques spécifiques à Managed File Transfer.

Utilisation de processus Connect:Direct pour soumettre des demandes de transfert Managed File Transfer

Vous pouvez soumettre une demande de transfert à l'agent de pont Connect:Direct à partir d'un processus Connect:Direct . Managed File Transfer fournit des commandes qui peuvent être appelées à partir d'une instruction **RUN TASK** dans un processus Connect:Direct .

Managed File Transfer fournit les commandes suivantes à utiliser avec les processus Connect:Direct :

ftetag

Spécifiez cette commande dans une étape qui précède la commande **ftebxf** ou **ftecxfer** pour créer les informations d'audit requises pour le transfert. Cette commande prend la spécification source du transfert comme paramètre. Pour plus d'informations sur le format de la spécification source, voir [«fteCreateTransfer: démarrer un nouveau transfert de fichier», à la page 447](#).

ftebxf

Indiquez cette commande pour créer une demande de transfert de fichier lorsque le gestionnaire de files d'attente auquel la demande de transfert est soumise se trouve sur le même système que le noeud Connect:Direct qui soumet la commande. Cette commande utilise les mêmes paramètres que la commande **fteCreateTransfer**. Pour plus d'informations sur ces paramètres, voir [«fteCreateTransfer: démarrer un nouveau transfert de fichier», à la page 447](#). Cette commande comporte également un paramètre supplémentaire:

-qmgrname

Obligatoire. Nom du gestionnaire de files d'attente auquel la commande doit être soumise.

ftecxfer

Indiquez cette commande pour créer une demande de transfert de fichier lorsque le gestionnaire de files d'attente auquel la demande de transfert est soumise se trouve sur un système différent du noeud Connect:Direct qui soumet la commande. Cette commande utilise les mêmes paramètres que la commande **fteCreateTransfer**. Pour plus d'informations sur les paramètres, voir [«fteCreateTransfer: démarrer un nouveau transfert de fichier», à la page 447](#). Cette commande comporte également trois paramètres supplémentaires:

-qmgrname

Obligatoire. Nom du gestionnaire de files d'attente auquel la commande doit être soumise.

-conname

Obligatoire. Hôte et port du gestionnaire de files d'attente sur lequel la commande doit être soumise, spécifié au format IBM MQ CONNAME. Par exemple, `host.example.com(1337)`.

-channelname

Facultatif. Nom du canal à utiliser pour la connexion au gestionnaire de files d'attente auquel la commande doit être soumise. S'il n'est pas spécifié, la valeur par défaut est `SYSTEM.DEF.SVRCONN` est utilisé.

Tâches associées

[«Création et soumission d'un processus Connect:Direct qui appelle Managed File Transfer à l'aide de Connect:Direct Requester», à la page 250](#)

Le demandeur Connect:Direct est une interface graphique que vous pouvez utiliser pour créer et soumettre un processus Connect:Direct qui appelle Managed File Transfer.

Référence associée

[«Exemple: un fichier de processus Connect:Direct qui appelle des commandes MFT», à la page 684](#)
Exemple de fichier de processus Connect:Direct qui appelle la commande Managed File Transfer **ftetag** et la commande **ftecxfer**.

Création et soumission d'un processus Connect:Direct qui appelle Managed File Transfer à l'aide de Connect:Direct Requester

Le demandeur Connect:Direct est une interface graphique que vous pouvez utiliser pour créer et soumettre un processus Connect:Direct qui appelle Managed File Transfer.

Pourquoi et quand exécuter cette tâche

Cette tâche explique comment créer un processus Connect:Direct qui appelle la commande Managed File Transfer **ftecxfer** ou la commande **ftebxf**. Utilisez la commande **ftecxfer** lorsque le gestionnaire de files d'attente auquel la demande de transfert est soumise se trouve sur un système différent du noeud Connect:Direct qui soumet la commande. Utilisez la commande **ftebxf** lorsque le gestionnaire de files d'attente auquel la demande de transfert est soumise se trouve sur le même système que le noeud Connect:Direct qui soumet la commande. La commande **ftecxfer** établit une connexion client au gestionnaire de files d'attente de l'agent source du transfert. Avant d'appeler la commande **ftecxfer**,

vous devez appeler la commande **ftetag** et lui transmettre les informations de spécification source. Cela permet au processus d'être consigné et audité de la même manière que les transferts lancés à partir de Managed File Transfer.

Procédure

1. Démarrez le demandeur Connect:Direct .
2. Dans l'onglet **Noeuds** du panneau, sélectionnez le noeud Connect:Direct utilisé comme noeud principal du processus.
3. Sélectionnez **Fichier > Nouveau > processus**. La fenêtre **Propriétés de processus** s'ouvre.
4. Dans la zone **Nom:** , entrez le nom du processus.
5. Sélectionnez le noeud secondaire dans la liste **Snode > Nom:** .
6. Sélectionnez le système d'exploitation du noeud secondaire dans la liste **Snode > Système d'exploitation:** .
7. Facultatif : Renseignez les informations supplémentaires de cette fenêtre dont vous avez besoin.
8. Cliquez sur **OK**. La fenêtre **Propriétés de processus** se ferme.
9. Créez une instruction qui exécute la commande Managed File Transfer **ftetag** .
 - a) Cliquez avec le bouton droit de la souris dans la fenêtre **Processus** de l'instruction **Fin** .
 - b) Sélectionnez **Insérer > Exécuter la tâche**. La fenêtre **Exécuter l'instruction de tâche** s'ouvre.
 - c) Dans la zone **Libellé:** , entrez Tag.
 - d) Dans la zone **Paramètres ou commandes facultatifs** , entrez `pgm(MQ_INSTALLATION_PATH/bin/ftetag) args(source_specification)`. Pour plus d'informations sur le format de *source_specification*, voir [«fteCreateTransfer: démarrer un nouveau transfert de fichier»](#), à la page 447.
 - e) Cliquez sur **OK**. La fenêtre **Exécuter l'instruction de tâche** se ferme.
10. Créez une instruction qui exécute la commande Managed File Transfer **ftecxfer** ou **ftebxfer** .
 - a) Cliquez avec le bouton droit de la souris dans la fenêtre **Processus** de l'instruction **Fin** .
 - b) Sélectionnez **Insérer > Exécuter la tâche**. La fenêtre **Exécuter l'instruction de tâche** s'ouvre.
 - c) Dans la zone **Libellé:** , entrez Transfer.
 - d) Dans la zone **Paramètres ou commandes facultatifs** , entrez `pgm(MQ_INSTALLATION_PATH/bin/ftecxfer) args(parameters)` ou `pgm(MQ_INSTALLATION_PATH/bin/ftebxfer) args(parameters)` en fonction de la commande que vous choisissez. Les paramètres utilisés par les commandes **ftecxfer** et **ftebxfer** sont les mêmes que ceux utilisés par la commande **fteCreateTransfer** , ainsi que certains paramètres supplémentaires spécifiques à **ftecxfer** et **ftebxfer**. Pour plus d'informations, voir [«fteCreateTransfer: démarrer un nouveau transfert de fichier»](#), à la page 447 et [«Utilisation de processus Connect:Direct pour soumettre des demandes de transfert Managed File Transfer»](#), à la page 249.
 - e) Cliquez sur **OK**. La fenêtre **Exécuter l'instruction de tâche** se ferme.
11. Facultatif : Créez les instructions supplémentaires dont vous avez besoin.
12. Soumettez le processus.
 - a) Cliquez avec le bouton droit de la souris dans la fenêtre **Processus** .
 - b) Sélectionnez **Soumettre**. La fenêtre **Connect:Direct Attach** s'ouvre.
 - c) Entrez le nom d'utilisateur et le mot de passe à utiliser pour exécuter le processus.
 - d) Cliquez sur **OK**.

Concepts associés

[«Utilisation de processus Connect:Direct pour soumettre des demandes de transfert Managed File Transfer»](#), à la page 249

Vous pouvez soumettre une demande de transfert à l'agent de pont Connect:Direct à partir d'un processus Connect:Direct . Managed File Transfer fournit des commandes qui peuvent être appelées à partir d'une instruction **RUN TASK** dans un processus Connect:Direct .

Utilisation de MFT à partir de IBM Integration Bus

Vous pouvez utiliser Managed File Transfer depuis IBM Integration Bus à l'aide des noeuds FTEOutput et FTEInput.

- Utilisez le noeud FTEInput pour transférer un fichier sur le réseau à l'aide de Managed File Transfer, puis traiter ce fichier dans le cadre d'un flux Integration Bus.
- Utilisez le noeud FTEOutput pour transférer un fichier généré par un flux Integration Bus à un autre emplacement sur le réseau.

Les agents qui transfèrent des fichiers vers ou depuis l'agent courtier peuvent se trouver à n'importe quel niveau de Managed File Transfer.

Pour plus d'informations, voir la [documentation du produit IBM Integration Bus](#).

MFT Reprise et redémarrage

Si votre agent ou votre gestionnaire de files d'attente n'est pas disponible pour une raison quelconque, par exemple en raison d'une panne d'alimentation ou de réseau, Managed File Transfer effectue la récupération suivante dans les scénarios suivants:

- Généralement, en cas de problème lors du transfert d'un fichier, Managed File Transfer récupère et redémarre ce transfert de fichier une fois le problème résolu.
- Si un fichier qui était en cours de transfert est supprimé ou modifié alors que l'agent ou le gestionnaire de files d'attente n'est pas disponible, le transfert échoue et vous obtenez un message dans le journal de transfert qui fournit des détails sur l'échec.
- Si un processus d'agent échoue lors d'un transfert de fichier, le transfert se poursuit lorsque vous redémarrez l'agent.
- Si un agent perd la connexion à son gestionnaire de files d'attente d'agent, l'agent attend en tentant de se reconnecter au gestionnaire de files d'attente. Lorsque l'agent se reconnecte à son gestionnaire de files d'attente, le transfert en cours se poursuit.
- Si l'agent est arrêté pour une raison quelconque, tous les moniteurs de ressources associés à une interrogation d'arrêt d'agent. Lorsque l'agent est restauré, les moniteurs sont également redémarrés et l'interrogation des ressources reprend.
- Pour un transfert de fichier avec une disposition source delete, si une récupération se produit après l'envoi de toutes les données d'un agent source à un agent cible, le fichier source est déverrouillé avant la suppression. Ce déverrouillage signifie que le fichier source peut être modifié avant la suppression du fichier. Par conséquent, il est considéré comme dangereux de supprimer le fichier source et l'avertissement suivant s'affiche:

```
BFGTR0075W: The source file has not been deleted because it is possible that the source file was modified after the source file was transferred.
```

Dans ce cas, vérifiez que le contenu du fichier source n'est pas modifié, puis supprimez manuellement le fichier source.

Vous pouvez vérifier le statut de vos transferts dans le IBM MQ Explorer. Si des transferts apparaissent sous la forme Stalled, vous devrez peut-être prendre des mesures correctives car le statut bloqué indique un problème avec l'agent ou entre les deux agents impliqués dans le transfert.

Concepts associés

«[Option de délai d'attente pour les transferts de fichiers en cours de récupération](#)», à la page 253

Vous pouvez définir la durée, en secondes, pendant laquelle un agent source tente de récupérer un transfert de fichiers bloqué. Si le transfert n'aboutit pas lorsque l'agent atteint le délai d'attente pour l'intervalle entre les nouvelles tentatives, le transfert échoue.

Option de délai d'attente pour les transferts de fichiers en cours de récupération

Vous pouvez définir la durée, en secondes, pendant laquelle un agent source tente de récupérer un transfert de fichiers bloqué. Si le transfert n'aboutit pas lorsque l'agent atteint le délai d'attente pour l'intervalle entre les nouvelles tentatives, le transfert échoue.

Avant IBM MQ 9.0.1, le comportement par défaut d'un agent source Managed File Transfer consiste à continuer de tenter de récupérer un transfert bloqué jusqu'à ce qu'il aboutisse. Le nouveau paramètre étant facultatif, vous n'avez pas besoin de le définir et les transferts suivent le comportement par défaut.

Depuis IBM MQ 9.0.1, vous pouvez spécifier l'une des options suivantes:

-1

L'agent continue de tenter de récupérer le transfert bloqué jusqu'à ce que le transfert aboutisse. Cette option est équivalente au comportement par défaut de l'agent si la propriété n'est pas définie.

0

L'agent arrête le transfert de fichiers dès qu'il entre dans la phase de reprise.

>0

L'agent continue de tenter de reprendre le transfert bloqué pendant le nombre de secondes défini par l'entier positif spécifié. La valeur 21600 indique que l'agent tente de récupérer le transfert pendant 6 heures à partir du moment où il passe à la récupération. La valeur maximale de ce paramètre est 99999999.

Définition du délai d'attente des nouvelles tentatives de reprise pour tous les transferts d'un agent source

Pour définir un délai de récupération qui s'applique à tous les transferts d'un agent source, ajoutez le paramètre et la paire de valeurs au fichier [agent.properties](#).

Dans cet exemple, la définition d'une valeur de délai d'attente de récupération de transfert sur -1 signifie que l'agent continue de tenter de récupérer le transfert bloqué jusqu'à ce qu'il aboutisse:

```
transferRecoveryTimeout=-1
```

La définition d'une valeur de délai d'attente de récupération de transfert sur 0 signifie que l'agent marque le transfert comme ayant échoué immédiatement après la saisie de la récupération:

```
transferRecoveryTimeout=0
```

La définition d'une valeur de délai de récupération de transfert de 21600 signifie que l'agent continue de relancer un transfert bloqué pendant 6 heures avant que le transfert ne soit marqué comme ayant échoué:

```
transferRecoveryTimeout=21600
```

Définition ou remplacement du délai d'attente des nouvelles tentatives de reprise pour les transferts individuels

Vous pouvez définir le paramètre de délai d'attente des nouvelles tentatives de reprise pour un transfert individuel lorsque vous:

- Création d'un transfert à l'aide de la commande [fteCreateTransfer](#) dans l'interface de ligne de commande Managed File Transfer ou IBM MQ Explorer
- Création d'un modèle de transfert à l'aide de la commande [«fteCreateTemplate: créer un nouveau modèle de transfert de fichiers»](#), à la page 431 dans l'interface de ligne de commande Managed File Transfer ou IBM MQ Explorer
- Création d'un moniteur à l'aide de l'interface de ligne de commande [fteCreateMonitor](#) Managed File Transfer ou IBM MQ Explorer

- Copie ou déplacement de fichiers à l'aide de tâches Ant `fte:filecopy` ou `fte:filemove`

La définition de la valeur `-rt` sur `-1` pour un transfert est l'équivalent du comportement par défaut, la récupération se poursuit jusqu'à ce que le transfert aboutisse ou qu'il soit annulé manuellement par l'utilisateur, par exemple

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -rt -1 -df C:\import\transferredfile.txt
C:\export\originalfile.txt
```

Pour plus d'informations sur l'utilisation du paramètre de délai d'attente de reprise de transfert lors de la création d'un nouveau transfert, voir [fteCreateTransfer command](#).

La définition de la valeur du paramètre `-rt` sur `0` indique que si le transfert initié à l'aide de ce modèle est bloqué, il échoue immédiatement et aucune reprise n'est tentée, par exemple

```
fteCreateTemplate -tn "payroll accounts monthly report template" -rt 0 -sa PAYROLL -sm
QM_PAYROLL1 -da ACCOUNTS
-dm QM_ACCOUNTS -df C:\payroll_reports\*.xls C:\out\*.xls
```

Pour plus d'informations sur l'utilisation du paramètre de délai d'attente de reprise de transfert lors de la création d'un modèle de transfert, voir [CommandefteCreateTemplate](#).

Vous pouvez utiliser IBM MQ Explorer pour définir le paramètre de délai de récupération et la valeur des transferts. Pour plus d'informations sur l'utilisation de IBM MQ Explorer pour configurer des transferts, voir «Démarrage d'un nouveau transfert de fichiers», à la page 163 et «Création d'un modèle de transfert de fichiers à l'aide de IBM MQ Explorer», à la page 202.

Vous pouvez également définir le délai d'attente de récupération en incluant l'option et la valeur **transferRecoveryTimeout**, avec les éléments **fte:filecopy** ou **fte:filemove** pour le déplacement ou la copie de fichiers à l'aide de tâches Ant, par exemple

```
<fte:filecopy cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
src="agent1@qm1" dst="agent2@qm2"
rcproperty="copy.result" transferRecoveryTimeout="0">
  <fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/file.bin"/>
</fte:filecopy>
```

Lorsque la tâche de copie de fichier est lancée et que le transfert entre en phase de récupération, le transfert s'arrête immédiatement sans tenter de récupération. La définition de l'option **transferRecoveryTimeout** avec **fte:filecopy** ou **fte:filemove** remplace la valeur définie dans le fichier `agent.properties`. Si la valeur **transferRecoveryTimeout** n'est pas définie avec **fte:filecopy** ou **fte:filemove**, la valeur du paramètre **transferRecoveryTimeout** du fichier `agent.properties` est utilisée. Pour plus d'informations, voir [«fte: tâche Ant filecopy»](#), à la page 878 et [«fte: tâche Ant filemove»](#), à la page 881.

Traitement de la priorité du délai de reprise

La valeur du délai d'attente de récupération de transfert, telle qu'elle est spécifiée via l'argument de l'interface de ligne de commande pour les commandes de création de transfert, de modèle ou de contrôle (y compris la définition de l'option dans l'assistant IBM MQ Explorer) ou comme spécifiée dans l'élément imbriqué **fte:filespec**, est prioritaire sur la valeur spécifiée pour le paramètre **transferRecoveryTimeout** dans le fichier `agent.properties` de l'agent source. Par exemple, pour la commande

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -df C:\import\transferredfile.txt
C:\export\originalfile.txt
```

qui est démarré sans le paramètre `-rt` et la paire de valeurs, l'agent source AGENT1 recherche une valeur **transferRecoveryTimeout** dans le fichier `agent.properties` afin de déterminer le comportement du délai d'attente de reprise.

Si dans le fichier `agent.properties`, **transferRecoveryTimeout** n'est pas défini ou est défini sur `-1`, l'agent suit le comportement par défaut et tente de récupérer le transfert jusqu'à ce qu'il aboutisse.

Lorsque l'option de délai de récupération **-rt** est spécifiée via l'interface de ligne de commande Managed File Transfer , par exemple, avec la commande **fteCreateTransfer** , cette valeur est prioritaire sur la valeur du fichier `agent.properties` et est utilisée comme paramètre pour le transfert.

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -rt 21600 -df C:\import\transferredfile.txt  
C:\export\originalfile.txt
```

Gestion du compteur de délai d'attente de reprise

Le compteur de délai d'attente de récupération démarre lorsque le transfert passe à l'état de récupération. Un message de journal de transfert est publié dans SYSTEM.FTE avec la chaîne de rubrique `Log/agent_name/transfer_ID` pour indiquer que le statut de transfert est remplacé par Récupération et l'heure d'horloge de l'agent source à laquelle le statut a été modifié. Si le transfert est repris dans l'intervalle de relance défini et n'atteint pas le délai de récupération (compteur <=délai de récupération), le compteur est réinitialisé à 0, prêt à être redémarré si le transfert entre en récupération.

Si le compteur atteint la valeur maximale définie pour le délai de récupération (compteur == délai de récupération), la récupération du transfert s'arrête et l'agent source signale que le transfert a échoué. Ce type d'échec de transfert, causé par le fait que le transfert a atteint le délai de récupération, est indiqué par un nouveau code de message, `RECOVERY_TIMEOUT (69)`. Un autre message de journal de transfert est publié dans SYSTEM.FTE , avec la chaîne de rubrique `Log/agent_name/transfer_ID`, pour indiquer que le transfert a échoué et inclut un nouveau message, le nouveau code retour et le journal des événements de l'agent source. Le journal des événements de l'agent source est mis à jour avec un message lorsque l'un des événements suivants se produit lors de la reprise:

- Lorsque le paramètre de délai d'attente de récupération est défini sur une valeur supérieure à -1, Managed File Transfer entre en phase de récupération. Le journal des événements de l'agent est mis à jour pour indiquer le début du temporisateur de récupération pour le **TransferId** et la durée pendant laquelle l'agent source attend avant de lancer le traitement du délai de récupération.
- Lors de la reprise de la récupération Managed File Transfer , le journal des événements de l'agent source est mis à jour avec un nouveau message pour indiquer que le **TransferId** qui était en cours de récupération est repris.
- Lorsqu'une récupération Managed File Transfer a dépassé le délai d'attente, le journal des événements de l'agent source est mis à jour pour indiquer le **TransferId** qui a échoué lors de la récupération, en raison du délai d'attente de récupération.

Ces messages de journal permettent aux utilisateurs (abonnés et consignateurs) d'identifier les transferts qui ont échoué en raison du délai d'attente de reprise de transfert.

Le compteur du délai de récupération est toujours au niveau de l'agent source. Toutefois, si l'agent de destination ne parvient pas à recevoir les informations de l'agent source dans les délais impartis, il peut envoyer une demande à l'agent source pour mettre le transfert en récupération. Pour un transfert où l'option de délai de récupération est définie, l'agent source démarre le compteur de délai de récupération lorsqu'il reçoit la demande de l'agent de destination.

Une gestion manuelle est toujours nécessaire pour les transferts qui n'utilisent pas l'option de délai de récupération, les transferts ayant échoué et partiellement terminés.

Pour les ensembles de transfert, où une seule demande de transfert est émise pour plusieurs fichiers et que certains des fichiers se sont terminés avec succès mais qu'un seul ne s'est terminé que partiellement, le transfert est toujours marqué comme ayant échoué car il ne s'est pas terminé comme prévu. Il se peut que l'agent source ait dépassé le délai d'attente lors du transfert du fichier partiellement terminé.

Vérifiez que l'agent de destination et le serveur de fichiers sont prêts et à l'état permettant d'accepter les transferts de fichiers.

Vous devez émettre à nouveau la demande de transfert pour l'ensemble complet, mais pour éviter les problèmes car certains des fichiers restent sur la destination à partir de la tentative de transfert initiale, vous pouvez émettre la nouvelle demande avec l'option `overwrite if existing` spécifiée. Cela permet de s'assurer que l'ensemble incomplet de fichiers de la tentative de transfert précédente est

nettoyé dans le cadre du nouveau transfert, avant que les fichiers ne soient à nouveau écrits dans la destination.

Traces et messages

Les points de trace sont inclus à des fins de diagnostic. La valeur du délai d'attente de reprise, le début de l'intervalle entre les nouvelles tentatives, le début de la période de reprise et la réinitialisation du compteur, et si le transfert a expiré ou a échoué, sont consignés. En cas de problème ou de comportement inattendu, vous pouvez collecter le journal de sortie et les fichiers de trace de l'agent source et les fournir à la demande du support IBM pour vous aider à résoudre les problèmes.

Les messages notifient à l'utilisateur qu'un transfert entre dans la reprise (BFGTR0081I) et qu'il est arrêté car il a dépassé le délai d'attente de la reprise (BFGSS0081E) et qu'il reprend après avoir été dans la reprise (BFGTR0082I).

BFGTR0001 - BFGTR9999

BFGTR0081I

BFGTR0082I

BFGSS0001 - BFGSS9999

BFGSS0081E

Développement d'applications pour MFT

Spécification des programmes à exécuter avec MFT

Vous pouvez exécuter des programmes sur un système sur lequel un Managed File Transfer Agent est en cours d'exécution. Dans le cadre d'une demande de transfert de fichiers, vous pouvez spécifier un programme à exécuter avant ou après le début d'un transfert. En outre, vous pouvez démarrer un programme qui ne fait pas partie d'une demande de transfert de fichier en soumettant une demande d'appel géré.

Il existe cinq scénarios dans lesquels vous pouvez spécifier un programme à exécuter:

- Dans le cadre d'une demande de transfert, au niveau de l'agent source, avant le début du transfert
- Dans le cadre d'une demande de transfert, au niveau de l'agent de destination, avant le début du transfert
- Dans le cadre d'une demande de transfert, au niveau de l'agent source, une fois le transfert terminé
- Dans le cadre d'une demande de transfert, au niveau de l'agent de destination, une fois le transfert terminé
- Pas dans le cadre d'une demande de transfert. Vous pouvez soumettre une demande à un agent pour exécuter un programme. Ce scénario est parfois appelé appel géré.

Les exits utilisateur et les appels de programme sont appelés dans l'ordre suivant:

```
- SourceTransferStartExit(onSourceTransferStart).  
- PRE_SOURCE Command.  
- DestinationTransferStartExits(onDestinationTransferStart).  
- PRE_DESTINATION Command.  
- The Transfer request is performed.  
- DestinationTransferEndExits(onDestinationTransferEnd).  
- POST_DESTINATION Command.  
- SourceTransferEndExits(onSourceTransferEnd).  
- POST_SOURCE Command.
```

Remarques :

1. Le **DestinationTransferEndExits** est exécuté uniquement lorsque le transfert est terminé, soit avec succès, soit partiellement.
2. Le **postDestinationCall** est exécuté uniquement lorsque le transfert est terminé, soit avec succès, soit partiellement.

3. Le **SourceTransferEndExits** est exécuté pour les transferts réussis, partiellement réussis ou ayant échoué.
4. **postSourceCall** est appelé uniquement si:
 - Le transfert n'a pas été annulé.
 - Le résultat est positif ou partiellement positif.
 - Tous les programmes de transfert post-destination ont été exécutés avec succès.

Il existe plusieurs façons de spécifier un programme que vous souhaitez exécuter. Ces options sont les suivantes :

Utiliser une tâche Ant Apache

Utilisez l'une des tâches Ant `fte:filecopy`, `fte:filemove` et `fte:call` pour démarrer un programme. A l'aide d'une tâche Ant, vous pouvez spécifier un programme dans l'un des cinq scénarios, à l'aide des éléments imbriqués `fte:presrc`, `fte:predst`, `fte:postdst`, `fte:postsrct` et `fte:command`. Pour plus d'informations, voir la section [«Éléments imbriqués d'appel de programme»](#), à la page 894.

Editer le message de demande de transfert de fichier

Vous pouvez éditer le code XML généré par une demande de transfert. Avec cette méthode, vous pouvez exécuter un programme dans l'un des cinq scénarios, en ajoutant des éléments **preSourceCall**, **postSourceCall**, **preDestinationCall**, **postDestinationCall** et **managedCall** au fichier XML. Utilisez ensuite ce fichier XML modifié comme définition de transfert pour une nouvelle demande de transfert de fichier, par exemple avec le paramètre **fteCreateTransfer -td**. Pour plus d'informations, voir [«Exemples de message de demande d'appel d'agent MFT»](#), à la page 820.

Utilisez la commande fteCreateTransfer

Vous pouvez utiliser la commande **fteCreateTransfer** pour spécifier les programmes à démarrer. Vous pouvez utiliser la commande pour spécifier des programmes à exécuter dans les quatre premiers scénarios, dans le cadre d'une demande de transfert, mais vous ne pouvez pas démarrer un appel géré. Pour plus d'informations sur les paramètres à utiliser, voir [«fteCreateTransfer: démarrer un nouveau transfert de fichier»](#), à la page 447. Pour des exemples d'utilisation de cette commande, voir [«Exemples d'utilisation de fteCreateTransfer pour démarrer des programmes»](#), à la page 869.

Référence associée

[«Propriété commandPath MFT»](#), à la page 374

La propriété `commandPath` permet de spécifier les emplacements à partir desquels Managed File Transfer peut exécuter des commandes. Soyez extrêmement prudent lorsque vous définissez cette propriété car toute commande de l'un des emplacements `commandPaths` spécifiés peut être effectivement appelée à partir d'un système client distant qui peut envoyer des commandes à l'agent.

Appels gérés

Les agents Managed File Transfer (MFT) sont généralement utilisés pour transférer des fichiers ou des messages. Ces transferts sont appelés *transferts gérés*. Les agents peuvent également être utilisés pour exécuter des commandes, des scripts ou des JCL sans qu'il soit nécessaire de transférer des fichiers ou des messages. Cette fonction est appelée *Appels gérés*.

Les demandes d'appels gérés peuvent être soumises à un agent de plusieurs manières:

- A l'aide de `fte: call` Ant task.
- Configuration d'un moniteur de ressources avec une tâche XML qui exécute une commande ou un script. Pour plus d'informations, voir [Configuration des tâches de surveillance pour lancer des commandes et des scripts](#).
- Insertion directe d'un message XML dans la file d'attente de commandes de l'agent. Voir [Format de message de demande de transfert de fichier](#) pour plus de détails sur le schéma XML d'appel géré.

Pour les appels gérés, le répertoire contenant la commande ou le script en cours d'exécution doit être spécifié dans la propriété d'agent **commandPath**.

Les appels gérés ne peuvent pas exécuter des commandes ou des scripts qui se trouvent dans des répertoires qui ne sont pas spécifiés dans le fichier **commandPath** de l'agent. Cela permet de s'assurer que l'agent n'exécute pas de code malveillant.

En outre, vous pouvez également activer la vérification des droits d'accès sur un agent pour vous assurer que seuls les utilisateurs autorisés sont autorisés à soumettre des demandes d'appels gérés. Pour plus d'informations à ce sujet, voir [Restriction des droits utilisateur sur les actions de l'agent MFT](#).

La commande, le script ou le JCL appelé dans le cadre d'un appel géré s'exécute en tant que processus externe, surveillé par l'agent. Lorsque le processus se termine, l'appel géré se termine et le code retour du processus est mis à la disposition de l'agent ou du script qui a appelé la tâche Ant **fte: Ant call**.

Si l'appel géré a été démarré par la tâche Ant **fte:call**, votre script Ant peut vérifier la valeur du code retour pour déterminer si l'appel géré a abouti ou non.

Pour tous les autres types d'appels gérés, vous pouvez spécifier les valeurs de code retour à utiliser pour indiquer que l'appel géré a abouti. L'agent compare le code retour du processus à ces codes retour lorsque le processus externe se termine.

Remarque : Etant donné que les appels gérés s'exécutent en tant que processus externes, ils ne peuvent pas être annulés une fois qu'ils ont démarré.

Appels gérés et emplacements de transfert source

Un agent contient un certain nombre d'emplacements de transfert source, comme indiqué par la propriété d'agent **maxSourceTransfers**, décrite dans [Propriétés d'agent avancées: Limite de transfert](#).

Chaque fois qu'un appel géré ou un transfert géré est exécuté, ils occupent un emplacement de transfert source. L'emplacement est libéré à la fin de l'appel géré ou du transfert géré.

Si tous les emplacements de transfert source sont utilisés lorsqu'un agent reçoit un nouvel appel géré ou une demande de transfert géré, la demande est mise en file d'attente par l'agent jusqu'à ce qu'un emplacement soit disponible.

Si un appel géré démarre un transfert géré (par exemple, si un appel géré exécute un script Ant et que ce script Ant utilise la tâche [fte: filecopy](#) ou [fte: filemove](#) pour transférer un fichier), deux emplacements de transfert source sont requis:

- Un pour le transfert géré
- Un pour l'appel géré

Dans cette situation, il est important de noter que si le transfert géré prend beaucoup de temps à se terminer ou est en cours de reprise, les deux emplacements de transfert source sont occupés jusqu'à ce que le transfert géré soit terminé ou annulé.

Par conséquent, vous devez envisager la conception d'un appel géré pour vous assurer qu'il n'occupe pas les créneaux de transfert source pendant une longue période.

Utilisation d' Apache Ant avec MFT

Managed File Transfer fournit des tâches que vous pouvez utiliser pour intégrer la fonction de transfert de fichier dans l'outil Ant Apache .

Vous pouvez utiliser la commande **fteAnt** pour exécuter des tâches Ant dans un environnement Managed File Transfer que vous avez déjà configuré. Vous pouvez utiliser des tâches Ant de transfert de fichiers à partir de vos scripts Ant pour coordonner des opérations de transfert de fichiers complexes à partir d'un langage de script interprété.

Pour plus d'informations sur Apache Ant, voir la page Web du projet Ant Apache : <https://ant.apache.org/>

Concepts associés

«Initiation à l'utilisation de scripts Ant avec MFT», à la page 259

L'utilisation de scripts Ant avec Managed File Transfer vous permet de coordonner des opérations de transfert de fichiers complexes à partir d'un langage de script interprété.

Référence associée

«Tâches Ant fournies par MFT», à la page 873

Managed File Transfer fournit un certain nombre de tâches Ant que vous pouvez utiliser pour accéder aux fonctions de transfert de fichiers.

«fteAnt: exécuter des tâches Ant dans MFT», à la page 389

La commande **fteAnt** exécute des scripts Ant dans un environnement dans lequel des tâches Ant Managed File Transfer sont disponibles.

«Exemples de tâches Ant pour MFT», à la page 260

Un certain nombre d'exemples de script Ant sont fournis avec votre installation de Managed File Transfer. Ces exemples se trouvent dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/fteant`. Chaque exemple de script contient une cible `init`, éditez les propriétés définies dans la cible `init` pour exécuter ces scripts avec votre configuration.

Initiation à l'utilisation de scripts Ant avec MFT

L'utilisation de scripts Ant avec Managed File Transfer vous permet de coordonner des opérations de transfert de fichiers complexes à partir d'un langage de script interprété.

Scripts Ant

Les scripts Ant (ou fichiers de génération) sont des documents XML qui définissent une ou plusieurs cibles. Ces cibles contiennent des éléments de tâche à exécuter. Managed File Transfer fournit des tâches que vous pouvez utiliser pour intégrer la fonction de transfert de fichier dans Apache Ant. Pour en savoir plus sur les scripts Ant, consultez la page Web du projet Ant Apache : <https://ant.apache.org/>

Des exemples de scripts Ant qui utilisent des tâches Managed File Transfer sont fournis avec l'installation de votre produit dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/fteant`

Sur les agents de pont de protocole, les scripts Ant sont exécutés sur le système d'agent de pont de protocole. Ces scripts Ant n'ont pas d'accès direct aux fichiers sur le serveur FTP ou SFTP.

Espace de nom

Un espace de nom est utilisé pour différencier les tâches Ant de transfert de fichier des autres tâches Ant qui peuvent partager le même nom. Vous définissez l'espace de nom dans la balise de projet de votre script Ant.

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns:fte="antlib:com.ibm.wmqfte.ant.taskdefs" default="do_ping">

  <target name="do_ping">
    <fte:ping cmdqm="qm@localhost@1414@SYSTEM.DEF.SVRCONN" agent="agent1@qm1"
      rcproperty="ping.rc" timeout="15"/>
  </target>

</project>
```

L'attribut `xmlns:fte="antlib:com.ibm.wmqfte.ant.taskdefs"` indique à Ant de rechercher les définitions des tâches préfixées par `fte` dans la bibliothèque `com.ibm.wmqfte.ant.taskdefs`.

Vous n'avez pas besoin d'utiliser `fte` comme préfixe d'espace de nom ; vous pouvez utiliser n'importe quelle valeur. Le préfixe d'espace de nom `fte` est utilisé dans tous les exemples et exemples de script Ant.

Exécution de scripts Ant

Pour exécuter des scripts Ant qui contiennent les tâches Ant de transfert de fichier, utilisez la commande **fteAnt** . Exemple :

```
fteAnt -file ant_script_location/ant_script_name
```

Pour plus d'informations, voir la section [«fteAnt: exécuter des tâches Ant dans MFT»](#), à la page 389.

Codes retour

Les tâches Ant de transfert de fichier renvoient les mêmes codes retour que les commandes Managed File Transfer . Pour plus d'informations, voir la section [«Codes retour pour MFT»](#), à la page 347.

Référence associée

[«Tâches Ant fournies par MFT»](#), à la page 873

Managed File Transfer fournit un certain nombre de tâches Ant que vous pouvez utiliser pour accéder aux fonctions de transfert de fichiers.

[«fteAnt: exécuter des tâches Ant dans MFT»](#), à la page 389

La commande **fteAnt** exécute des scripts Ant dans un environnement dans lequel des tâches Ant Managed File Transfer sont disponibles.

[«Exemples de tâches Ant pour MFT»](#), à la page 260

Un certain nombre d'exemples de script Ant sont fournis avec votre installation de Managed File Transfer. Ces exemples se trouvent dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/fteant`. Chaque exemple de script contient une cible `init` , éditez les propriétés définies dans la cible `init` pour exécuter ces scripts avec votre configuration.

Exemples de tâches Ant pour MFT

Un certain nombre d'exemples de script Ant sont fournis avec votre installation de Managed File Transfer. Ces exemples se trouvent dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/fteant`. Chaque exemple de script contient une cible `init` , éditez les propriétés définies dans la cible `init` pour exécuter ces scripts avec votre configuration.

adresse électronique

L'exemple email montre comment utiliser des tâches Ant pour transférer un fichier et envoyer un courrier électronique à une adresse électronique spécifiée en cas d'échec du transfert. Le script vérifie que les agents source et de destination sont actifs et qu'ils peuvent traiter les transferts à l'aide de la tâche Managed File Transfer `ping` . Si les deux agents sont actifs, le script utilise la tâche Managed File Transfer `filecopy` pour transférer un fichier entre les agents source et cible, sans supprimer le fichier d'origine. Si le transfert échoue, le script envoie un courrier électronique contenant des informations sur l'échec à l'aide de la tâche Ant `email standard`.

hub

L'exemple `hub` est constitué de deux scripts: `hubcopy.xml` et `hubprocess.xml` . Le script `hubcopy.xml` montre comment utiliser des scripts Ant pour générer des topologies de style 'hub and spoke'. Dans cet exemple, deux fichiers sont transférés des agents s'exécutant sur des machines satellites vers un agent s'exécutant sur la machine concentrateur. Les deux fichiers sont transférés en même temps et lorsque les transferts sont terminés, le script Ant `hubprocess.xml` est exécuté sur la machine du concentrateur pour traiter les fichiers. Si les deux fichiers sont transférés correctement, le script Ant concatène le contenu des fichiers. Si les fichiers ne sont pas transférés correctement, le script Ant nettoie en supprimant les données de fichier qui ont été transférées. Pour que cet exemple fonctionne correctement, vous devez placer le script `hubprocess.xml` sur le chemin de commande de l'agent concentrateur. Pour plus d'informations sur la définition du chemin de commande d'un agent, voir [commandPath](#).

librarytransfer (plateforme IBM i uniquement)

IBM i

IBM i L'exemple `librarytransfer` montre comment utiliser des tâches Ant pour transférer une bibliothèque IBM i sur un système IBM i vers un deuxième système IBM i .

IBM i

IBM WebSphere MQ File Transfer Edition 7.0.2 on IBM i n'inclut pas la prise en charge directe des transferts d'objets de bibliothèque IBM i native. L'exemple `librarytransfer` utilise la prise en charge des fichiers de sauvegarde natifs sous IBM i avec des tâches Ant prédéfinies disponibles dans Managed File Transfer pour transférer des objets de bibliothèque natifs entre deux systèmes IBM i . L'exemple utilise un élément imbriqué `<presrc>` dans une tâche de `filecopy` Managed File Transfer pour appeler un script exécutable `librarysave.sh` qui sauvegarde la bibliothèque demandée sur le système de l'agent source dans un fichier de sauvegarde temporaire. Le fichier sauvegarde est déplacé par la tâche Ant `filecopy` vers le système de l'agent cible où un élément imbriqué `<postdst>` est utilisé pour appeler le script exécutable `libraryrestore.sh` afin de restaurer la bibliothèque sauvegardée dans le fichier sauvegarde sur le système cible.

IBM i

Avant d'exécuter cet exemple, vous devez effectuer une configuration comme décrit dans le fichier `librarytransfer.xml` . Vous devez également disposer d'un environnement Managed File Transfer opérationnel sur deux machines IBM i . La configuration doit se composer d'un agent source s'exécutant sur la première machine IBM i et d'un agent cible s'exécutant sur la deuxième machine IBM i . Les deux agents doivent pouvoir communiquer entre eux.

IBM i

L'exemple `librarytransfer` comprend les trois fichiers suivants:

- `librarytransfer.xml`
- `librarysave.sh` (script exécutable `<presrc>`)
- `libraryrestore.sh` (script exécutable `<postdst>`)

Les exemples de fichiers se trouvent dans le répertoire suivant: `/QIBM/ProdData/WMQFTE/V7/samples/fteant/ibmi/librarytransfer`

IBM i

Pour exécuter cet exemple, l'utilisateur doit effectuer les étapes suivantes:

1. Démarrez une session Qshell. Dans une fenêtre de commande IBM i , entrez: `STRQSH`
2. Accédez au répertoire `bin` comme suit:

```
cd /QIBM/ProdData/WMQFTE/V7/bin
```

3. Une fois la configuration requise terminée, exécutez l'exemple à l'aide de la commande suivante:

```
fteant -f /QIBM/ProdData/WMQFTE/V7/samples/fteant/ibmi/librarytransfer/librarytransfer.xml
```

physicalfiletransfer (plateforme IBM i uniquement)

IBM i

L'exemple `physicalfiletransfer` montre comment utiliser des tâches Ant pour transférer un fichier source physique ou de base de données d'une bibliothèque sur un système IBM i vers une bibliothèque sur un deuxième système IBM i .

IBM i

IBM WebSphere MQ File Transfer Edition 7.0.2 on IBM i n'inclut pas la prise en charge directe des transferts de fichiers source physiques ou de base de données natifs sur IBM i . L'exemple `physicalfiletransfer` utilise la prise en charge du fichier de sauvegarde natif sous IBM i avec des tâches Ant prédéfinies disponibles dans Managed File Transfer pour transférer des fichiers source physiques et de base de données complets entre deux systèmes IBM i . L'exemple utilise un élément imbriqué `<presrc>` dans une tâche Managed File Transfer `filecopy` pour appeler un script exécutable `physicalfilesave.sh` afin de sauvegarder le fichier source physique ou de base de données demandé

à partir d'une bibliothèque sur le système de l'agent source dans un fichier de sauvegarde temporaire. Le fichier sauvegarde est déplacé par la tâche Ant filecopy vers le système de l'agent cible où un élément imbriqué `< postdst>` est utilisé pour appeler le script exécutable `physicalfilerestore.sh`, puis restaure l'objet fichier dans le fichier sauvegarde dans une bibliothèque spécifiée sur le système cible.

IBM i Avant d'exécuter cet exemple, vous devez effectuer une configuration comme décrit dans le fichier `physicalfiletransfer.xml`. Vous devez également disposer d'un environnement Managed File Transfer opérationnel sur deux systèmes IBM i. La configuration doit se composer d'un agent source s'exécutant sur le premier système IBM i et d'un agent cible s'exécutant sur le second système IBM i. Les deux agents doivent pouvoir communiquer entre eux.

IBM i L'exemple `physicalfiletransfer` comprend les trois fichiers suivants:

- `physicalfiletransfer.xml`
- `physicalfilesave.sh` (script exécutable `< presrc>`)
- `physicalfilerestore.sh` (script exécutable `< postdst>`)

Les exemples de fichiers se trouvent dans le répertoire suivant: `/QIBM/ProdData/WMQFTE/V7/samples/fteant/ibmi/physicalfiletransfer`

IBM i Pour exécuter cet exemple, l'utilisateur doit effectuer les étapes suivantes:

1. Démarrez une session Qshell. Dans une fenêtre de commande IBM i, entrez: `STRQSH`
2. Accédez au répertoire `bin` comme suit:

```
cd /QIBM/ProdData/WMQFTE/V7/bin
```

3. Une fois la configuration requise terminée, exécutez l'exemple à l'aide de la commande suivante:

```
fteant -f /QIBM/ProdData/WMQFTE/V7/samples/fteant/ibmi/physicalfiletransfer/physicalfiletransfer.xml
```

délai d'attente

L'exemple `timeout` montre comment utiliser des tâches Ant pour tenter un transfert de fichier et annuler le transfert s'il dure plus longtemps qu'une valeur de délai d'attente spécifiée. Le script lance un transfert de fichier à l'aide de la tâche Managed File Transfer `filecopy`. Le résultat de ce transfert est différé. Le script utilise la tâche Managed File Transfer `<fte: tâche Ant awaitissue>`, à la page 873 pour attendre un nombre de secondes donné pour que le transfert se termine. Si le transfert n'est pas terminé dans le délai imparti, la tâche Managed File Transfer `<fte: annuler la tâche Ant>`, à la page 877 est utilisée pour annuler le transfert de fichier.

vsamtransfer

z/OS

z/OS L'exemple `vsamtransfer` montre comment utiliser des tâches Ant pour effectuer un transfert d'un fichier VSAM vers un autre fichier VSAM à l'aide de Managed File Transfer. Actuellement, Managed File Transfer ne prend pas en charge le transfert de fichiers VSAM. L'exemple de script décharge les enregistrements de données VSAM dans un fichier séquentiel à l'aide de l'élément imbriqué `presrc` `<Eléments imbriqués d'appel de programme>`, à la page 894 pour appeler le fichier exécutable `datasetcopy.sh`. Le script utilise la tâche Managed File Transfer `<fte: tâche Ant filemove>`, à la page 881 pour transférer le fichier séquentiel de l'agent source vers l'agent cible. Le script utilise ensuite l'élément imbriqué `postdst` `<Eléments imbriqués d'appel de programme>`, à la page 894 pour appeler le script `loadvsam.jcl`. Ce script JCL charge les enregistrements de fichier transférés dans un fichier VSAM de destination. Cet exemple utilise JCL pour l'appel de destination afin de présenter cette option de langage. Le même résultat peut également être obtenu en utilisant un second script shell à la place.

z/OS Cet exemple ne nécessite pas que les fichiers source et cible soient VSAM. L'exemple fonctionne pour tous les jeux de données si les jeux de données source et cible sont du même type.

z/OS Pour que cet exemple fonctionne correctement, vous devez placer le script `datasetcopy.sh` sur le chemin de commande de l'agent source et le script `loadvsam.jcl` sur le chemin de commande de l'agent cible. Pour plus d'informations sur la définition du chemin de commande d'un agent, voir [commandPath](#).

zip

L'exemple zip est constitué de deux scripts: `zip.xml` et `zipfiles.xml`. L'exemple montre comment utiliser l'élément imbriqué `presrc` «Eléments imbriqués d'appel de programme», à la page 894 dans la tâche Managed File Transfer «fte: tâche Ant filemove», à la page 881 pour exécuter un script Ant avant d'effectuer une opération de déplacement de transfert de fichier. Le script `zipfiles.xml` appelé par l'élément imbriqué `presrc` dans le script `zip.xml` compresse le contenu d'un répertoire. Le script `zip.xml` transfère le fichier compressé. Cet exemple nécessite que le script Ant `zipfiles.xml` soit présent dans le chemin de commande de l'agent source. En effet, le script Ant `zipfiles.xml` contient la cible utilisée pour compresser le contenu du répertoire sur l'agent source. Pour plus d'informations sur la définition du chemin de commande d'un agent, voir [commandPath](#).

Personnalisation de MFT avec des exits utilisateur

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Managed File Transfer fournit des points dans le code où Managed File Transfer peut transmettre le contrôle à un programme que vous avez écrit (une routine d'exit utilisateur). Ces points sont appelés points d'exit utilisateur. Managed File Transfer peut alors reprendre le contrôle lorsque votre programme a terminé son travail. Vous n'avez pas besoin d'utiliser les exits utilisateur, mais ils sont utiles si vous souhaitez étendre et personnaliser la fonction de votre système Managed File Transfer pour répondre à vos besoins spécifiques.

Il existe deux points au cours du traitement du transfert de fichiers où vous pouvez appeler un exit utilisateur sur le système source et deux points au cours du traitement du transfert de fichiers où vous pouvez appeler un exit utilisateur sur le système cible. Le tableau suivant récapitule chacun de ces points d'exit utilisateur et l'interface Java que vous devez implémenter pour utiliser les points d'exit.

Point de sortie	Interface Java à implémenter
Points d'exit côté source:	
Avant le début du transfert de fichiers complet	SourceTransferStartExit.java
Une fois le transfert de fichiers complet terminé	SourceTransferEndExit.java
Points d'exit côté destination:	
Avant le début du transfert de fichiers complet	DestinationTransferStartExit.java
Une fois le transfert de fichiers complet terminé	DestinationTransferEndExit.java

Les exits utilisateur sont appelés dans l'ordre suivant:

1. `SourceTransferStartExit`
2. `DestinationTransferStartExit`
3. `DestinationTransferEndExit`
4. `SourceTransferEndExit`

Les modifications apportées par les exits `SourceTransferStartExit` et `DestinationTransferStartExit` sont propagées en tant qu'entrées aux exits suivants. Par exemple, si l'exit `SourceTransferStartExit` modifie les métadonnées de transfert, les modifications sont reflétées dans les métadonnées de transfert d'entrée vers les autres exits.

Les exits utilisateur et les appels de programme sont appelés dans l'ordre suivant:

```
- SourceTransferStartExit(onSourceTransferStart).
- PRE_SOURCE Command.
- DestinationTransferStartExits(onDestinationTransferStart).
- PRE_DESTINATION Command.
- The Transfer request is performed.
- DestinationTransferEndExits(onDestinationTransferEnd).
- POST_DESTINATION Command.
- SourceTransferEndExits(onSourceTransferEnd).
- POST_SOURCE Command.
```

Remarques :

1. Le **DestinationTransferEndExits** est exécuté uniquement lorsque le transfert est terminé, soit avec succès, soit partiellement.
2. Le **postDestinationCall** est exécuté uniquement lorsque le transfert est terminé, soit avec succès, soit partiellement.
3. Le **SourceTransferEndExits** est exécuté pour les transferts réussis, partiellement réussis ou ayant échoué.
4. **postSourceCall** est appelé uniquement si:
 - Le transfert n'a pas été annulé.
 - Le résultat est positif ou partiellement positif.
 - Tous les programmes de transfert post-destination ont été exécutés avec succès.

Génération de votre exit utilisateur

Les interfaces permettant de générer un exit utilisateur sont contenues dans `MQ_INSTALL_DIRECTORY/mqft/lib/com.ibm.wmqfte.exitroutines.api.jar`. Vous devez inclure ce fichier .jar dans le chemin d'accès aux classes lorsque vous générez votre exit. Pour exécuter l'exit, extrayez l'exit en tant que fichier .jar et placez ce fichier .jar dans un répertoire, comme décrit dans la section suivante.

Emplacements d'exit utilisateur

Vous pouvez stocker vos routines d'exit utilisateur dans deux emplacements possibles:

- Le répertoire `exits`. Il existe un répertoire d'exits sous chaque répertoire d'agent. Par exemple :
`var\mqm\mqft\config\QM_JUPITER\agents\AGENT1\exits`
- Vous pouvez définir la propriété de chemin `exitClass` pour spécifier un autre emplacement. S'il existe des classes d'exit dans le répertoire `exits` et dans le chemin d'accès aux classes défini par le chemin d'accès `exitClass`, les classes du répertoire `exits` sont prioritaires, ce qui signifie que s'il existe des classes dans les deux emplacements portant le même nom, les classes du répertoire `exits` sont prioritaires.

Configuration d'un agent pour l'utilisation des exits utilisateur

Quatre propriétés d'agent peuvent être définies pour spécifier les exits utilisateur qu'un agent appelle. Ces propriétés d'agent sont `sourceTransferStartExitClasses`, `sourceTransferEndExitClasses`, `destinationTransferStartExitClasses` et `destinationTransferEndExitClasses`. Pour plus d'informations sur l'utilisation de ces propriétés, voir [«Propriétés de l'agent MFT pour les exits utilisateur»](#), à la page 906.

Exécution des exits utilisateur sur les agents de pont de protocole

Lorsque l'agent source appelle l'exit, il transmet à l'exit une liste des éléments source pour le transfert. Pour les agents normaux, il s'agit d'une liste de noms de fichiers complets. Comme les fichiers doivent être locaux (ou accessibles via un montage), l'exit peut y accéder et les chiffrer.

Toutefois, pour un agent de pont de protocole, les entrées de la liste sont au format suivant:

```
"<file server identifiant>:<fully-qualified file name of the file on the remote file server>"
```

Pour chaque entrée de la liste, l'exit doit d'abord se connecter au serveur de fichiers (via FTP). protocoles FTPS ou SFTP), téléchargez le fichier, chiffrez-le localement, puis téléchargez le fichier chiffré sur le serveur de fichiers.

Exécution des exits utilisateur sur les agents de pont Connect:Direct

Vous ne pouvez pas exécuter d'exits utilisateur sur des agents de pont Connect:Direct .

Exits utilisateur source et cible MFT

Séparateurs de répertoire

Les séparateurs de répertoire dans les spécifications de fichier source sont toujours représentés à l'aide de barres obliques (/), quelle que soit la façon dont vous avez spécifié les séparateurs de répertoire dans la commande **fteCreateTransfer** ou dans IBM MQ Explorer. Vous devez en tenir compte lorsque vous écrivez un exit. Par exemple, si vous souhaitez vérifier que le fichier source suivant existe: c : \a\b .txt et que vous avez spécifié ce fichier source à l'aide de la commande **fteCreateTransfer** ou de IBM MQ Explorer, notez que le nom de fichier est stocké sous la forme suivante: c : /a/b .txt Donc, si vous recherchez la chaîne d'origine de c : \a\b .txt, vous ne trouverez pas de correspondance.

Points d'exit côté source

Avant le début du transfert de fichiers complet

Cet exit est appelé par l'agent source lorsqu'une demande de transfert est suivante dans la liste des transferts en attente et que le transfert est sur le point de démarrer.

Par exemple, vous pouvez utiliser ce point d'exit pour envoyer des fichiers par étapes dans un répertoire auquel l'agent a accès en lecture / écriture à l'aide d'une commande externe ou pour renommer les fichiers sur le système cible.

Transmettez les arguments suivants à cet exit:

- Nom de l'agent source
- Nom de l'agent de destination
- Métadonnées d'environnement
- Métadonnées de transfert
- Spécifications de fichier (y compris les métadonnées de fichier)

Les données renvoyées par cet exit sont les suivantes:

- Métadonnées de transfert mises à jour. Les entrées peuvent être ajoutées, modifiées et supprimées.
- Liste mise à jour des spécifications de fichier, qui se compose de paires de nom de fichier source et de nom de fichier cible. Les entrées peuvent être ajoutées, modifiées et supprimées
- Indicateur qui indique si le transfert doit être poursuivi
- Chaîne à insérer dans le journal de transfert.

Implémentez l'interface [SourceTransferStartExit.java](#) pour appeler le code d'exit utilisateur à ce point d'exit.

Une fois le transfert de fichiers complet terminé

Cet exit est appelé par l'agent source une fois le transfert de fichier complet terminé.

Un exemple d'utilisation de ce point d'exit consiste à effectuer des tâches d'achèvement, telles que l'envoi d'un courrier électronique ou d'un message IBM MQ pour indiquer que le transfert est terminé.

Transmettez les arguments suivants à cet exit:

- Résultat de l'exit de transfert
- Nom de l'agent source
- Nom de l'agent de destination
- Métadonnées d'environnement
- Métadonnées de transfert
- Résultats du fichier

Les données renvoyées par cet exit sont les suivantes:

- Chaîne mise à jour à insérer dans le journal de transfert.

Implémentez l'interface [SourceTransferEndExit.java](#) pour appeler le code d'exit utilisateur à ce point d'exit.

Points d'exit côté destination

Avant le début du transfert de fichiers complet

Un exemple d'utilisation de ce point d'exit consiste à valider les droits d'accès à la destination.

Transmettez les arguments suivants à cet exit:

- Nom de l'agent source
- Nom de l'agent de destination
- Métadonnées d'environnement
- Métadonnées de transfert
- Spécifications de fichier

Les données renvoyées par cet exit sont les suivantes:

- Ensemble de noms de fichiers de destination mis à jour. Les entrées peuvent être modifiées mais ne peuvent pas être ajoutées ou supprimées.
- Indicateur qui indique si le transfert doit être poursuivi
- Chaîne à insérer dans le journal de transfert.

Implémentez l'interface [DestinationTransferStartExit.java](#) pour appeler le code d'exit utilisateur à ce point d'exit.

Une fois le transfert de fichiers complet terminé

Un exemple d'utilisation de cet exit utilisateur consiste à démarrer un traitement par lots qui utilise les fichiers transférés ou à envoyer un courrier électronique si le transfert a échoué.

Transmettez les arguments suivants à cet exit:

- Résultat de l'exit de transfert
- Nom de l'agent source
- Nom de l'agent de destination
- Métadonnées d'environnement
- Métadonnées de transfert
- Résultats du fichier

Les données renvoyées par cet exit sont les suivantes:

- Chaîne mise à jour à insérer dans le journal de transfert.

Implémentez l'interface `DestinationTransferEndExit.java` pour appeler le code d'exit utilisateur à ce point d'exit.

Concepts associés

«Interfaces Java pour les exits utilisateur MFT», à la page 908

Utilisez les rubriques de cette section pour obtenir des informations de référence sur les interfaces Java pour les routines d'exit utilisateur.

Référence associée

«Activation du débogage à distance pour les exits utilisateur MFT», à la page 270

Lorsque vous développez vos exits utilisateur, vous pouvez utiliser un débogueur pour localiser les problèmes dans votre code.

«Exemple d'exit utilisateur de transfert de source MFT», à la page 270

«Exits utilisateur du moniteur de ressources MFT», à la page 902

Les exits utilisateur du moniteur de ressources permettent de configurer le code personnalisé à exécuter lorsque la condition de déclenchement d'un moniteur est remplie, avant le démarrage de la tâche associée.

Utilisation des exits utilisateur d'E-S de transfert MFT

Vous pouvez utiliser les exits utilisateur d'E-S de transfert Managed File Transfer pour configurer le code personnalisé afin d'effectuer le travail d'E-S de système de fichiers sous-jacent pour les transferts Managed File Transfer .

Généralement, pour les transferts MFT , un agent effectue une sélection parmi l'un des fournisseurs d'E-S intégrés pour interagir avec les systèmes de fichiers appropriés pour le transfert. Les fournisseurs d'E-S intégrés prennent en charge les types de système de fichiers suivants:

- Systèmes de fichiers UNIX-type et Windows-type standard
-  Fichiers séquentiels et partitionnés z/OS (sous z/OS uniquement)
-  Fichiers de sauvegarde natifs IBM i (sous IBM i uniquement)
- Files d'attente IBM MQ
- Serveurs de protocole FTP et SFTP distants (pour les agents de pont de protocole uniquement)
- Noeuds Connect:Direct distants (pour les agents de pont Connect:Direct uniquement)

Pour les systèmes de fichiers qui ne sont pas pris en charge ou pour lesquels vous avez besoin d'un comportement d'E-S personnalisé, vous pouvez écrire un exit utilisateur d'E-S de transfert.

Les exits utilisateur d'E-S de transfert utilisent l'infrastructure existante pour les exits utilisateur. Toutefois, ces exits utilisateur d'E-S de transfert diffèrent des autres exits utilisateur car leur fonction est accessible plusieurs fois tout au long du transfert pour chaque fichier.

Utilisez la propriété d'agent `IOExitClasses` (dans le fichier `agent.properties`) pour spécifier les classes d'exit d'E-S à charger. Séparez chaque classe d'exit par une virgule, par exemple:

```
IOExitClasses=testExits.TestExit1,testExits.testExit2
```

Les interfaces Java pour les exits utilisateur d'E-S de transfert sont les suivantes:

Exit d'E-S

Point d'entrée principal utilisé pour déterminer si l'exit d'E-S est utilisé. Cette instance est chargée de créer des instances `IOExitPath` .

Vous devez spécifier uniquement l'interface d'exit d'E-S `IOExit` pour la propriété d'agent `IOExitClasses`.

IOExitPath

Représente une interface abstraite ; par exemple, un conteneur de données ou un caractère générique représentant un ensemble de conteneurs de données. Vous ne pouvez pas créer une instance de classe qui implémente cette interface. L'interface permet d'examiner le chemin et de répertorier les chemins dérivés. Les interfaces `IOExitResourcePath` et `IOExitWildcardPath` étendent `IOExitPath`.

IOExitChannel

Permet de lire ou d'écrire des données dans une ressource `IOExitPath`.

Canal IOExitRecord

Étend l'interface `IOExitChannel` pour les ressources `IOExitPath` orientées enregistrement, ce qui permet de lire ou d'écrire des données dans une ressource `IOExitPath` en multiples d'enregistrements.

IOExitLock

Représente un verrou sur une ressource `IOExitPath` pour un accès partagé ou exclusif.

z/OS IOExitRecordResourcePath

Étend l'interface de chemin `IOExitResourcePath` pour représenter un conteneur de données pour un fichier orienté enregistrement ; par exemple, un fichier `z/OS`. Vous pouvez utiliser l'interface pour localiser des données et créer des instances de canal `IOExitRecord` pour des opérations de lecture ou d'écriture.

Chemin d'accès à IOExitResource

Étend l'interface `IOExitPath` pour représenter un conteneur de données ; par exemple, un fichier ou un répertoire. Vous pouvez utiliser l'interface pour localiser des données. Si l'interface représente un répertoire, vous pouvez utiliser la méthode `listPaths` pour renvoyer une liste de chemins.

Chemin IOExitWildcard

Étend l'interface `IOExitPath` pour représenter un chemin indiquant un caractère générique. Vous pouvez utiliser cette interface pour faire correspondre plusieurs chemins `IOExitResource`.

IOExitProperties

Indique les propriétés qui déterminent comment Managed File Transfer gère `IOExitPath` pour certains aspects des E-S. Par exemple, s'il faut utiliser des fichiers intermédiaires ou relire une ressource depuis le début si un transfert est redémarré.

IBM i Exemples d'exit utilisateur MFT on IBM i

Managed File Transfer fournit des exemples d'exit utilisateur spécifiques à IBM i avec votre installation. Les exemples se trouvent dans les répertoires `MQMFT_install_dir/samples/ioexit-IBMi` et `MQMFT_install_dir/samples/userexit-IBMi`.

com.ibm.wmqfte.exit.io.ibm.i.qdls.FTEQDLSExit

L'exemple d'exit utilisateur `com.ibm.wmqfte.exit.io.ibm.i.qdls.FTEQDLSExit` transfère des fichiers dans le système de fichiers QDLS sous IBM i. Une fois l'exit installé, tous les transferts vers des fichiers qui commencent par `/QDLS` utilisent automatiquement l'exit.

Pour installer cet exit, procédez comme suit:

1. Copiez le fichier `com.ibm.wmqfte.samples.ibm.i.ioexits.jar` du répertoire `WMQFTE_install_dir/samples/ioexit-IBMi` vers le répertoire `exits` de l'agent.
2. Ajoutez `com.ibm.wmqfte.exit.io.ibm.i.qdls.FTEQDLSExit` à la propriété `IOExitClasses`.
3. Redémarrez l'agent.

com.ibm.wmqfte.exit.user.ibm1.FileMemberMonitorExit

L'exemple d'exit utilisateur `com.ibm.wmqfte.exit.user.ibm1.FileMemberMonitorExit` se comporte comme un moniteur de fichiers MFT et transfère automatiquement des membres de fichiers physiques à partir d'une bibliothèque IBM i.

Pour exécuter cet exit, spécifiez une valeur pour la zone de métadonnées "library.qsys.monitor" (à l'aide du paramètre `-md`, par exemple). Ce paramètre prend un chemin de style IFS vers un membre de fichier et peut contenir des caractères génériques de fichier et de membre. Par exemple, `/QSYS.LIB/FOO.LIB/BAR.FILE/*.MBR`, `/QSYS.LIB/FOO.LIB/*.FILE/BAR.MBR`, `/QSYS.LIB/FOO.LIB/*.FILE/*`. Région d'équilibrage de charge.

Cet exemple d'exit comporte également une zone de métadonnées facultative "naming.scheme.qsys.monitor", que vous pouvez utiliser pour déterminer le schéma de dénomination utilisé lors du transfert. Par défaut, cette zone est définie sur "unix", ce qui entraîne le nom du fichier de destination `FOO.MBR`. Vous pouvez également spécifier la valeur "ibmi" pour utiliser le fichier IBM i FTP `FILE.MEMBER`, par exemple, `/QSYS.LIB/FOO.LIB/BAR.FILE/BAZ.MBR` est transféré en tant que fichier d'archives du courtier `BAR.BAZ`.

Pour installer cet exit, procédez comme suit:

1. Copiez le fichier `com.ibm.wmqfte.samples.ibm1.userexits.jar` du répertoire `WMQFTE_install_dir/samples/userexit-IBM1` vers le répertoire `exits` de l'agent.
2. Ajoutez `com.ibm.wmqfte.exit.user.ibm1.FileMemberMonitorExit` à la propriété `sourceTransferStartExitClasses` dans le fichier `agent.properties`.
3. Redémarrez l'agent.

com.ibm.wmqfte.exit.user.ibm1.EmptyFileDeleteExit

L'exemple d'exit utilisateur `com.ibm.wmqfte.exit.user.ibm1.EmptyFileDeleteExit` supprime un objet fichier vide lorsque le membre de fichier source est supprimé dans le cadre du transfert. Étant donné que les objets de fichier IBM i peuvent potentiellement contenir de nombreux membres, les objets de fichier sont traités comme des répertoires par MFT. Par conséquent, vous ne pouvez pas effectuer d'opération de déplacement sur un objet fichier à l'aide de MFT; les opérations de déplacement sont prises en charge au niveau du membre uniquement. Par conséquent, lorsque vous effectuez une opération de déplacement sur un membre, le fichier désormais vide est laissé derrière. Utilisez cet exemple d'exit si vous souhaitez supprimer ces fichiers vides dans le cadre de la demande de transfert.

Si vous spécifiez "true" pour les métadonnées "empty.file.delete" et que vous transférez un `FTEFileMember`, l'exemple d'exit supprime le fichier parent si le fichier est vide.

Pour installer cet exit, procédez comme suit:

1. Copiez le fichier `com.ibm.wmqfte.samples.ibm1.userexits.jar` depuis `WMQFTE_install_dir/samples/userexit-IBM1` dans le répertoire `exits` de l'agent.
2. Ajoutez `com.ibm.wmqfte.exit.user.ibm1.EmptyFileDeleteExit` à la propriété `sourceTransferStartExitClasses` dans le fichier `agent.properties`.
3. Redémarrez l'agent.

Référence associée

«Utilisation des exits utilisateur d'E-S de transfert MFT», à la page 267

Vous pouvez utiliser les exits utilisateur d'E-S de transfert Managed File Transfer pour configurer le code personnalisé afin d'effectuer le travail d'E-S de système de fichiers sous-jacent pour les transferts Managed File Transfer.

«Propriétés de l'agent MFT pour les exits utilisateur», à la page 906

Outre les propriétés standard du fichier `agent.properties`, il existe plusieurs propriétés avancées spécifiques aux routines d'exit utilisateur. Ces propriétés ne sont pas incluses par défaut. Par conséquent, si vous souhaitez les utiliser, vous devez éditer manuellement le fichier `agent.properties`. Si vous modifiez le fichier `agent.properties` alors que cet agent est en cours d'exécution, arrêtez et redémarrez l'agent pour récupérer les modifications.

Activation du débogage à distance pour les exits utilisateur MFT

Lorsque vous développez vos exits utilisateur, vous pouvez utiliser un débogueur pour localiser les problèmes dans votre code.

Etant donné que les exits s'exécutent dans la machine virtuelle Java qui exécute l'agent, vous ne pouvez pas utiliser le support de débogage direct qui est généralement inclus dans un environnement de développement intégré. Toutefois, vous pouvez activer le débogage à distance de la machine virtuelle Java, puis connecter un débogueur à distance approprié.

Pour activer le débogage à distance, utilisez les paramètres JVM standard **-Xdebug** et **-Xrunjdwp**. Ces propriétés sont transmises à la machine virtuelle Java qui exécute l'agent par la variable d'environnement **BFG_JVM_PROPERTIES**. Par exemple, sous UNIX, les commandes suivantes démarrent l'agent et entraînent la machine virtuelle Java à écouter les connexions du débogueur sur le port TCP 8765.

```
export BFG_JVM_PROPERTIES="-Xdebug -Xrunjdwp:transport=dt_socket,server=y,address=8765"
fteStartAgent -F TEST_AGENT
```

L'agent ne démarre pas tant que le débogueur ne se connecte pas. Utilisez la commande **set** sous Windows à la place de la commande **export**.

Vous pouvez également utiliser d'autres méthodes de communication entre le débogueur et la machine virtuelle Java. Par exemple, la machine virtuelle Java peut ouvrir la connexion au débogueur au lieu de l'inverse, ou vous pouvez utiliser la mémoire partagée au lieu de TCP. Pour plus de détails, voir la documentation [Java Platform Debugger Architecture](#).

Vous devez utiliser le paramètre **-F** (avant-plan) lorsque vous démarrez l'agent en mode débogage à distance.

Utilisation du débogueur Eclipse

Les étapes suivantes s'appliquent à la fonction de débogage à distance dans l'environnement de développement Eclipse. Vous pouvez également utiliser d'autres débogueurs distants compatibles avec JPDA.

1. Cliquez sur **Exécuter** > **Ouvrir la boîte de dialogue de débogage** (ou sur **Exécuter** > **Configurations de débogage** ou sur **Exécuter** > **Boîte de dialogue de débogage** en fonction de votre version d'Eclipse).
2. Cliquez deux fois sur **Application Java distante** dans la liste des types de configuration pour créer une configuration de débogage.
3. Renseignez les zones de configuration et sauvegardez la configuration de débogage. Si vous avez déjà démarré la machine virtuelle Java de l'agent en mode débogage, vous pouvez vous connecter à la machine virtuelle Java maintenant.

Exemple d'exit utilisateur de transfert de source MFT

```
/*
 * A Sample Source Transfer End Exit that prints information about a transfer to standard
 * output.
 * If the agent is run in the background the output will be sent to the agent's event log file.
 * If
 * the agent is started in the foreground by specifying the -F parameter on the fteStartAgent
 * command the output will be sent to the console.
 *
 * To run the exit execute the following steps:
 *
 * Compile and build the exit into a jar file. You need the following in the class path:
 * {MQ_INSTALLATION_PATH}\mqft\lib\com.ibm.wmqfte.exitroutines.api.jar
 *
 * Put the jar in your agent's exits directory:
 * {MQ_DATA_PATH}\config\coordQmgrName\agents\agentName\exits\
 */
```

```

* Update the agent's properties file:
* {MQ_DATA_PATH}\config\coordQmgrName\agents\agentName\agent.properties
* to include the following property:
* sourceTransferEndExitClasses=[packageName.]SampleEndExit
*
* Restart agent to pick up the exit
*
* Send the agent a transfer request:
* For example: fteCreateTransfer -sa myAgent -da YourAgent -df output.txt input.txt
*/

import java.util.List;
import java.util.Map;
import java.util.Iterator;

import com.ibm.wmqfte.exitroutine.api.SourceTransferEndExit;
import com.ibm.wmqfte.exitroutine.api.TransferExitResult;
import com.ibm.wmqfte.exitroutine.api.FileTransferResult;

public class SampleEndExit implements SourceTransferEndExit {

    public String onSourceTransferEnd(TransferExitResult transferExitResult,
        String sourceAgentName,
        String destinationAgentName,
        Map<String, String>environmentMetaData,
        Map<String, String>transferMetaData,
        List<FileTransferResult>fileResults) {

        System.out.println("Environment Meta Data: " + environmentMetaData);
        System.out.println("Transfer Meta Data: " + transferMetaData);

        System.out.println("Source agent: " +
            sourceAgentName);
        System.out.println("Destination agent: " +
            destinationAgentName);

        if (fileResults.isEmpty()) {
            System.out.println("No files in the list");
            return "No files";
        }
        else {

            System.out.println( "File list: ");

            final Iterator<FileTransferResult> iterator = fileResults.iterator();

            while (iterator.hasNext()){
                final FileTransferResult thisFileSpec = iterator.next();
                System.out.println("Source file spec: " +
                    thisFileSpec.getSourceFileSpecification() +
                    ", Destination file spec: " +
                    thisFileSpec.getDestinationFileSpecification());
            }
            return "Done";
        }
    }
}

```

Exemple d'exit utilisateur de données d'identification de pont de protocole

Pour plus d'informations sur l'utilisation de cet exemple d'exit utilisateur, voir [«Mappage des données d'identification pour un serveur de fichiers à l'aide de classes d'exit»](#), à la page 230

```

import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.util.Enumeration;
import java.util.HashMap;
import java.util.Map;
import java.util.Properties;
import java.util.StringTokenizer;

import com.ibm.wmqfte.exitroutine.api.CredentialExitResult;

```

```

import com.ibm.wmqfte.exitroutine.api.CredentialExitResultCode;
import com.ibm.wmqfte.exitroutine.api.CredentialPassword;
import com.ibm.wmqfte.exitroutine.api.CredentialUserId;
import com.ibm.wmqfte.exitroutine.api.Credentials;
import com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit;

/**
 * A sample protocol bridge credential exit
 *
 * This exit reads a properties file that maps mq user ids to server user ids
 * and server passwords. The format of each entry in the properties file is:
 *
 * mqUserId=serverUserId,serverPassword
 *
 * The location of the properties file is taken from the protocol bridge agent
 * property protocolBridgeCredentialConfiguration.
 *
 * To install the sample exit compile the class and export to a jar file.
 * Place the jar file in the exits subdirectory of the agent data directory
 * of the protocol bridge agent on which the exit is to be installed.
 * In the agent.properties file of the protocol bridge agent set the
 * protocolBridgeCredentialExitClasses to SampleCredentialExit
 * Create a properties file that contains the mqUserId to serverUserId and
 * serverPassword mappings applicable to the agent. In the agent.properties
 * file of the protocol bridge agent set the protocolBridgeCredentialConfiguration
 * property to the absolute path name of this properties file.
 * To activate the changes stop and restart the protocol bridge agent.
 *
 * For further information on protocol bridge credential exits refer to
 * the WebSphere MQ Managed File Transfer documentation online at:
 * https://www.ibm.com/docs/SSEP7X_7.0.4/welcome/WelcomePagev7r0.html
 */
public class SampleCredentialExit implements ProtocolBridgeCredentialExit {

    // The map that holds mq user ID to serverUserId and serverPassword mappings
    final private Map<String,Credentials> credentialsMap = new HashMap<String, Credentials>();

    /* (non-Javadoc)
     * @see com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit#initialize(java.util.Map)
     */
    public synchronized boolean initialize(Map<String, String> bridgeProperties) {

        // Flag to indicate whether the exit has been successfully initialized or not
        boolean initialisationResult = true;

        // Get the path of the mq user ID mapping properties file
        final String propertiesFilePath = bridgeProperties.get("protocolBridgeCredentialConfiguration");

        if (propertiesFilePath == null || propertiesFilePath.length() == 0) {
            // The properties file path has not been specified. Output an error and return false
            System.err.println("Error initializing SampleCredentialExit.");
            System.err.println("The location of the mqUserId mapping properties file has not been
specified in the
protocolBridgeCredentialConfiguration property");
            initialisationResult = false;
        }

        if (initialisationResult) {

            // The Properties object that holds mq user ID to serverUserId and serverPassword
            // mappings from the properties file
            final Properties mappingProperties = new Properties();

            // Open and load the properties from the properties file
            final File propertiesFile = new File (propertiesFilePath);
            FileInputStream inputStream = null;
            try {
                // Create a file input stream to the file
                inputStream = new FileInputStream(propertiesFile);

                // Load the properties from the file
                mappingProperties.load(inputStream);
            }
            catch (FileNotFoundException ex) {
                System.err.println("Error initializing SampleCredentialExit.");
                System.err.println("Unable to find the mqUserId mapping properties file: " +
propertiesFilePath);
                initialisationResult = false;
            }
            catch (IOException ex) {
                System.err.println("Error initializing SampleCredentialExit.");
                System.err.println("Error loading the properties from the mqUserId mapping properties

```

```

file: " + propertiesFilePath);
        initialisationResult = false;
    }
    finally {
        // Close the inputStream
        if (inputStream != null) {
            try {
                inputStream.close();
            }
            catch (IOException ex) {
                System.err.println("Error initializing SampleCredentialExit.");
                System.err.println("Error closing the mqUserId mapping properties file: " +
propertiesFilePath);
            }
        }
        initialisationResult = false;
    }
}

if (initialisationResult) {
    // Populate the map of mqUserId to server credentials from the properties
    final Enumeration<?> propertyNames = mappingProperties.propertyNames();
    while ( propertyNames.hasMoreElements()) {
        final Object name = propertyNames.nextElement();
        if (name instanceof String ) {
            final String mqUserId = ((String)name).trim();
            // Get the value and split into serverUserId and serverPassword
            final String value = mappingProperties.getProperty(mqUserId);
            final StringTokenizer valueTokenizer = new StringTokenizer(value, ",");
            String serverUserId = "";
            String serverPassword = "";
            if (valueTokenizer.hasMoreTokens()) {
                serverUserId = valueTokenizer.nextToken().trim();
            }
            if (valueTokenizer.hasMoreTokens()) {
                serverPassword = valueTokenizer.nextToken().trim();
            }
            // Create a Credential object from the serverUserId and serverPassword
            final Credentials credentials = new Credentials(new CredentialUserId(serverUserId), new
CredentialPassword(serverPassword));
            // Insert the credentials into the map
            credentialsMap.put(mqUserId, credentials);
        }
    }
}

return initialisationResult;
}
/* (non-Javadoc)
 * @see com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit#mapMQUserId(java.lang.String)
 */
public synchronized CredentialExitResult mapMQUserId(String mqUserId) {
    CredentialExitResult result = null;
    // Attempt to get the server credentials for the given mq user id
    final Credentials credentials = credentialsMap.get(mqUserId.trim());
    if ( credentials == null) {
        // No entry has been found so return no mapping found with no credentials
        result = new CredentialExitResult(CredentialExitResultCode.NO_MAPPING_FOUND, null);
    }
    else {
        // Some credentials have been found so return success to the user along with the credentials
        result = new CredentialExitResult(CredentialExitResultCode.USER_SUCCESSFULLY_MAPPED,
credentials);
    }
    return result;
}
/* (non-Javadoc)
 * @see com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit#shutdown(java.util.Map)
 */
public void shutdown(Map<String, String> bridgeProperties) {
    // Nothing to do in this method because there are no resources that need to be released
}
}

```

Exemple d'exit utilisateur de propriétés de pont de protocole

Pour plus d'informations sur l'utilisation de cet exemple d'exit utilisateur, voir «[ProtocolBridgePropertiesExit2: recherche des propriétés du serveur de fichiers de protocole](#)», à la page 224

SamplePropertiesExit2.java

```
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.util.HashMap;
import java.util.Map;
import java.util.Map.Entry;
import java.util.Properties;

import com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit2;
import com.ibm.wmqfte.exitroutine.api.ProtocolServerPropertyConstants;

/**
 * A sample protocol bridge properties exit. This exit reads a properties file
 * that contains properties for protocol servers.
 * <p>
 * The format of each entry in the properties file is:
 * {@literal serverName=type://host:port}
 * Ensure there is a default entry such as
 * {@literal default=type://host:port}
 * otherwise the agent will fail to start with a BFGBR0168 as it must have a
 * default server.
 * <p>
 * The location of the properties file is taken from the protocol bridge agent
 * property {@code protocolBridgePropertiesConfiguration}.
 * <p>
 * The methods {@code getCredentialLocation} returns the location of the associated
 * ProtocolBridgeCredentials.xml, this sample it is defined to be stored in a directory
 * defined by the environment variable CREDENTIALSHOME
 * <p>
 * To install the sample exit:
 * <ol>
 * <li>Compile the class and export to a jar file.
 * <li>Place the jar file in the {@code exits} subdirectory of the agent data directory
 * of the protocol bridge agent on which the exit is to be installed.
 * <li>In the {@code agent.properties} file of the protocol bridge agent
 * set the {@code protocolBridgePropertiesExitClasses} to
 * {@code SamplePropertiesExit2}.
 * <li>Create a properties file that contains the appropriate properties to specify the
 * required servers.
 * <li>In the {@code agent.properties} file of the protocol bridge agent
 * set the <code>protocolBridgePropertiesConfiguration</code> property to the
 * absolute path name of this properties file.
 * <li>To activate the changes stop and restart the protocol bridge agent.
 * </ol>
 * <p>
 * For further information on protocol bridge properties exits refer to the
 * WebSphere MQ Managed File Transfer documentation online at:
 * <p>
 * {@link https://www.ibm.com/docs/SSEP7X_7.0.4/welcome/WelcomePagev7r0.html}
 */
public class SamplePropertiesExit2 implements ProtocolBridgePropertiesExit2 {

    /**
     * Helper class to encapsulate protocol server information.
     */
    private static class ServerInformation {
        private final String type;
        private final String host;
        private final int port;

        public ServerInformation(String url) {
            int index = url.indexOf("://");
            if (index == -1) throw new IllegalArgumentException("Invalid server URL: "+url);
            type = url.substring(0, index);

            int portIndex = url.indexOf(":", index+3);
            if (portIndex == -1) {
                host = url.substring(index+3);
                port = -1;
            } else {
                host = url.substring(index+3, portIndex);
            }
        }
    }
}
```

```

        port = Integer.parseInt(url.substring(portIndex+1));
    }
}

public String getType() {
    return type;
}

public String getHost() {
    return host;
}

public int getPort() {
    return port;
}
}

/** A {@code Map} that holds information for each configured protocol server */
final private Map<String, ServerInformation> servers = new HashMap<String, ServerInformation>();

/* (non-Javadoc)
 * @see
 com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit#getProtocolServerProperties(java.lang.String)
 */
public Properties getProtocolServerProperties(String protocolServerName) {
    // Attempt to get the protocol server information for the given protocol server name
    // If no name has been supplied then this implies the default.
    final ServerInformation info;
    if (protocolServerName == null || protocolServerName.length() == 0) {
        protocolServerName = "default";
    }
    info = servers.get(protocolServerName);

    // Build the return set of properties from the collected protocol server information, when
available.
    // The properties set here is the minimal set of properties to be a valid set.
    final Properties result;
    if (info != null) {
        result = new Properties();
        result.setProperty(ProtocolServerPropertyConstants.SERVER_NAME, protocolServerName);
        result.setProperty(ProtocolServerPropertyConstants.SERVER_TYPE, info.getType());
        result.setProperty(ProtocolServerPropertyConstants.SERVER_HOST_NAME, info.getHost());
        if (info.getPort() != -1)
result.setProperty(ProtocolServerPropertyConstants.SERVER_PORT_VALUE, ""+info.getPort());
        result.setProperty(ProtocolServerPropertyConstants.SERVER_PLATFORM, "UNIX");
        if (info.getType().toUpperCase().startsWith("FTP")) { // FTP & FTPS
            result.setProperty(ProtocolServerPropertyConstants.SERVER_TIMEZONE, "Europe/London");
            result.setProperty(ProtocolServerPropertyConstants.SERVER_LOCALE, "en-GB");
        }
        result.setProperty(ProtocolServerPropertyConstants.SERVER_FILE_ENCODING, "UTF-8");
    } else {
        System.err.println("Error no default protocol file server entry has been supplied");
        result = null;
    }

    return result;
}

/* (non-Javadoc)
 * @see com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit#initialize(java.util.Map)
 */
public boolean initialize(Map<String, String> bridgeProperties) {
    // Flag to indicate whether the exit has been successfully initialized or not
    boolean initialisationResult = true;

    // Get the path of the properties file
    final String propertiesFilePath = bridgeProperties.get("protocolBridgePropertiesConfiguration");
    if (propertiesFilePath == null || propertiesFilePath.length() == 0) {
        // The protocol server properties file path has not been specified. Output an error and
return false
        System.err.println("Error initializing SamplePropertiesExit.");
        System.err.println("The location of the protocol server properties file has not been
specified in the
protocolBridgePropertiesConfiguration property");
        initialisationResult = false;
    }

    if (initialisationResult) {
        // The Properties object that holds protocol server information
        final Properties mappingProperties = new Properties();

```

```

// Open and load the properties from the properties file
final File propertiesFile = new File (propertiesFilePath);
FileInputStream inputStream = null;
try {
    // Create a file input stream to the file
    inputStream = new FileInputStream(propertiesFile);

    // Load the properties from the file
    mappingProperties.load(inputStream);
} catch (final FileNotFoundException ex) {
    System.err.println("Error initializing SamplePropertiesExit.");
    System.err.println("Unable to find the protocol server properties file: " +
propertiesFilePath);
    initialisationResult = false;
} catch (final IOException ex) {
    System.err.println("Error initializing SamplePropertiesExit.");
    System.err.println("Error loading the properties from the protocol server properties
file: " + propertiesFilePath);
    initialisationResult = false;
} finally {
    // Close the inputStream
    if (inputStream != null) {
        try {
            inputStream.close();
        } catch (final IOException ex) {
            System.err.println("Error initializing SamplePropertiesExit.");
            System.err.println("Error closing the protocol server properties file: " +
propertiesFilePath);
            initialisationResult = false;
        }
    }
}

if (initialisationResult) {
    // Populate the map of protocol servers from the properties
    for (Entry<Object, Object> entry : mappingProperties.entrySet()) {
        final String serverName = (String)entry.getKey();
        final ServerInformation info = new ServerInformation((String)entry.getValue());
        servers.put(serverName, info);
    }
}

return initialisationResult;
}

/* (non-Javadoc)
 * @see com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit#shutdown(java.util.Map)
 */
public void shutdown(Map<String, String> bridgeProperties) {
    // Nothing to do in this method because there are no resources that need to be released
}

/* (non-Javadoc)
 * @see com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit2#getCredentialLocation()
 */
public String getCredentialLocation() {
    String envLocationPath;
    if (System.getProperty("os.name").toLowerCase().contains("win")) {
        // Windows style
        envLocationPath = "%CREDENTIALSHOME%\\ProtocolBridgeCredentials.xml";
    }
    else {
        // Unix style
        envLocationPath = "$CREDENTIALSHOME/ProtocolBridgeCredentials.xml";
    }
    return envLocationPath;
}
}
}

```

Contrôle de MFT via le placement de messages dans la file d'attente de commandes d'agent

Vous pouvez écrire une application qui contrôle Managed File Transfer via le placement de messages dans des files d'attente de commandes d'agent.

Vous pouvez placer un message dans la file d'attente de commandes d'un agent pour demander que l'agent effectue l'une des actions suivantes :

- Créer un transfert de fichier
- Créer un transfert de fichier planifié
- Annuler un transfert de fichier
- Annuler un transfert de fichier planifié
- Appeler une commande
- Créer un moniteur
- Supprimer un moniteur
- Envoyer une commande ping pour vérifier que l'agent est actif

Pour que l'agent effectue l'une de ces actions, le message doit être dans un format XML conforme à l'un des schémas suivants :

FileTransfer.xsd

Vous pouvez utiliser des messages dans ce format pour créer un transfert de fichier ou un transfert de fichier planifié, appeler une commande, ou annuler un transfert de fichier ou un transfert de fichier planifié. Pour plus d'informations, voir la section [«Format de message de demande de transfert de fichier»](#), à la page 804.

Monitor.xsd

Vous pouvez utiliser des messages dans ce format pour créer ou supprimer un moniteur de ressources. Pour plus d'informations, voir la section [«Formats des messages de demande du moniteur MFT»](#), à la page 822.

PingAgent.xsd

Vous pouvez utiliser des messages dans ce format pour envoyer une commande ping à un agent afin de vérifier qu'il est actif. Pour plus d'informations, voir la section [«Format de message de demande de l'agent MFT Ping»](#), à la page 831.

L'agent renvoie une réponse aux messages de demande. Le message de réponse est placé dans une file d'attente de réponses qui est définie dans le message de demande. Son format est le format XML défini par le schéma suivant :

Reply.xsd

Pour plus d'informations, voir la section [«Format du message de réponse de l'agent MFT»](#), à la page 832.

traitement des incidentsMFT

Utilisez les informations de référence suivantes pour vous aider à diagnostiquer les erreurs dans Managed File Transfer:

MFT identification des problèmes généraux

Utilisez les informations de référence suivantes pour vous aider à diagnostiquer les erreurs dans Managed File Transfer:

Concepts associés

[«Conseils et astuces pour l'utilisation d' MFT»](#), à la page 279

Voici quelques suggestions pour vous aider à utiliser au mieux Managed File Transfer.

[«Conseils pour l'exécution d'un agent ou d'un consignateur MFT en tant que service Windows»](#), à la page 333

Vous pouvez exécuter un agent Managed File Transfer , un consignateur de base de données autonome et un consignateur de fichiers autonome, en tant que services Windows . Si vous rencontrez un problème avec ces services Windows , vous pouvez utiliser les fichiers journaux de service et les informations de cette rubrique pour diagnostiquer le problème.

Tâches associées

«Exécution de la trace sous MFT», à la page 280

Vous pouvez utiliser les méthodes décrites dans cette section pour tracer Managed File Transfer.

«Que faire si vous recevez une erreur lors de la mise à jour de votre schéma de base de données MFT sur une base de données Oracle», à la page 338

Vous pouvez recevoir le message d'erreur suivant lors de la mise à jour de votre schéma de base de données vers le niveau le plus récent à l'aide du fichier `ftelog_tables_oracle_702_703.sql` :
ERROR at line 1: ORA-02289: sequence does not exist. Cette erreur se produit car les séquences et les déclencheurs utilisés par les tables ne se trouvent pas dans le même schéma que les tables.

Référence associée

«Problèmes MFT courants», à la page 306

Problèmes courants susceptibles de se produire dans votre réseau Managed File Transfer .

«Que faire si votre agent MFT n'est pas répertorié par la commande `fteListAgents ?`», à la page 309

Si votre agent n'est pas répertorié par la commande **`fteListAgents`** ou n'est pas affiché dans IBM MQ Explorer, ou si vos transferts de fichier ne sont pas affichés dans le **journal de transfert** de IBM MQ Explorer, vous pouvez effectuer un certain nombre d'étapes afin d'identifier le problème.

«Que faire si votre processus d'agent MFT disparaît mais qu'aucune information de diagnostic n'est consignée», à la page 311

Sur les plateformes UNIX , si un processus d'agent a disparu mais que les fichiers journaux de l'agent ne contiennent aucune explication, cela peut être dû à la manière dont l'agent a été démarré.

«Que faire si vous pensez que votre transfert de fichiers est bloqué», à la page 321

Sur un système très chargé ou en cas de problèmes de réseau entre les agents source et cible, les transferts peuvent parfois sembler bloqués dans un état de file d'attente ou de récupération. Un certain nombre de facteurs peuvent être à l'origine de cette situation.

«Que faire si votre agent de pont de protocole signale qu'un fichier est introuvable», à la page 322

Lorsque l'agent de pont de protocole signale que le serveur SFTP ou FTP auquel se connecte le pont de protocole renvoie un message d'erreur `File not found` , ce message peut indiquer que l'un des nombreux cas d'erreur s'est produit.

«Que faire si les fichiers de destination créés par un transfert démarré par un moniteur de ressources de file d'attente contiennent des données incorrectes», à la page 324

Vous pouvez créer un moniteur de ressources pour surveiller une file d'attente et transférer un message ou un groupe de messages d'une file d'attente vers un fichier. Le nom de fichier peut être spécifié à l'aide des descripteurs de message MQMD du message ou du premier message d'un groupe. Si un transfert de message à fichier échoue et que le message ou le groupe est laissé dans la file d'attente, la prochaine fois que le moniteur est déclenché, il peut entraîner la création de fichiers contenant des données incorrectes.

«Que faire si des messages sont générés dans votre système SYSTEM.MANAGED.DURABLE files d'attente ou remplissage de votre système de fichiers», à la page 326

Si votre plug-in IBM MQ Explorer utilise un abonnement durable sur le gestionnaire de files d'attente de coordination, les messages peuvent s'accumuler dans 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.

«Examen des messages avant la publication», à la page 326

Etant donné que les agents peuvent se connecter aux gestionnaires de files d'attente IBM WebSphere MQ 6.0 , ils n'utilisent pas l'approche de publication directe introduite dans IBM WebSphere MQ 7.0 . A la place, les agents envoient des messages ordinaires au gestionnaire de files d'attente de coordination qui contient un en-tête MQRFH. L'en-tête MQRFH demande que le contenu du message soit publié. Ces messages sont envoyés à SYSTEM.FTE sur le gestionnaire de file d'attente de coordination et les messages sont généralement publiés immédiatement à partir de cette file d'attente. Si des conditions d'erreur arrêtent cette publication, vous pouvez examiner les messages de la file d'attente avant de tenter de la publier pour faciliter le diagnostic. Pour ce faire, procédez comme suit:

«Erreurs possibles lors du transfert des fichiers sauvegarde IBM i», à la page 328

Si vous utilisez Managed File Transfer pour transférer plusieurs fois le même fichier de sauvegarde IBM i , le transfert risque d'échouer.

«Conseils pour la définition des attributs MQ et des propriétés MFT associées à la taille des messages», à la page 329

Vous pouvez modifier les attributs IBM MQ et les propriétés Managed File Transfer pour affecter le comportement de Managed File Transfer lors de la lecture ou de l'écriture de messages de différentes tailles.

«Traitement et rejet des erreurs du consignateur MFT», à la page 339

Le consignateur Managed File Transfer identifie deux types d'erreur: les erreurs par message et les erreurs générales.

«Que faire si le consignateur MFT est démarré, mais qu'aucune information de transfert n'est consignée dans la base de données», à la page 340

Les tables de base de données utilisées par le consignateur Managed File Transfer requièrent que la taille de page de la base de données soit supérieure ou égale à 8 Ko. Si la taille de page de la base de données n'est pas assez grande, les tables ne sont pas créées correctement et l'erreur SQLSTATE=42704s'affiche.

«fteDisplayVersion», à la page 483

Utilisez la commande **fteDisplayVersion** pour afficher la version de Managed File Transfer que vous avez installée.

«Erreurs BFGSS0023E et comment les éviter», à la page 344

Si vous désinstallez un groupe de correctifs d'une installation afin de revenir à une version précédente du produit et qu'un agent associé à l'installation était impliqué dans des transferts gérés au moment de la désinstallation, cet agent ne peut pas démarrer et signale une erreur BFGSS0023E . Vous pouvez éviter cette erreur en effectuant un certain nombre d'étapes qui doivent empêcher l'affichage des messages BFGSS0023E lors du redémarrage des agents.

«Codes retour pour MFT», à la page 347

Les commandes Managed File Transfer , les tâches Ant et les messages de journal fournissent des codes retour indiquant si les fonctions ont abouti.

Conseils et astuces pour l'utilisation d' MFT

Voici quelques suggestions pour vous aider à utiliser au mieux Managed File Transfer.

- Si vous modifiez le fichier `agent.properties` , arrêtez et redémarrez l'agent pour appliquer les modifications.
- Si vous démarrez un transfert de fichiers et qu'il n'y a aucun signe de progression du transfert et qu'aucune erreur n'est signalée, vérifiez que l'agent source est en cours d'exécution. Si le transfert est affiché mais ne progresse pas, vérifiez que l'agent de destination est également en cours d'exécution. Vous pouvez vérifier l'état en cours des agents dans le journal de l'agent ou vérifier que l'agent est actif à l'aide d'une commande **ftePingAgent** .
- Lorsque vous annulez un transfert individuel à l'aide de la commande **fteCancelTransfer** , vous pouvez utiliser l'agent source ou de destination dans le paramètre **-agentName** . Toutefois, lorsque vous supprimez une planification de transfert à l'aide de la commande **fteDeleteScheduledTransfer** , vous devez utiliser le nom de l'agent source dans le paramètre **-agentName** .
- Lorsque vous créez un transfert de fichier, les chemins de fichier source et de destination, absolus ou relatifs, sont significatifs uniquement sur les agents source et de destination. Le système et le répertoire à partir duquel la commande **fteCreateAgent** est émise ne sont pas pertinents pour le fichier en cours de transfert.
- Il se peut que votre configuration d'environnement par défaut ne puisse pas prendre entièrement en charge Managed File Transfer, en particulier si vous exécutez plusieurs transferts simultanés. Si un agent présente une erreur indiquant qu'il manque de mémoire, vérifiez et mettez à jour les paramètres suivants, selon les besoins:
 -  Pour les plateformes UNIX : exécutez la commande: `ulimit -m 1048576` (ou environ 1 Go). Cette taille d'ensemble résidente maximale est suffisante pour permettre un maximum de 25

transferts simultanés (25 transferts simultanés est la valeur par défaut pour le nombre maximal de transferts pour un agent).

- Pour toutes les plateformes: définissez la variable d'environnement **BFG_JVM_PROPERTIES** comme suit: `BFG_JVM_PROPERTIES="-Xmx1024M"`

Si vous souhaitez autoriser un nombre de transferts simultanés supérieur à la valeur par défaut maximale de 25, utilisez des tailles pour **ulimit** et **BFG_JVM_PROPERTIES** supérieures à celles suggérées.

Remarque : Pour les agents de pont Connect:Direct , la valeur par défaut du nombre maximal de transferts simultanés est 5.

- Lorsque vous utilisez Managed File Transfer pour transférer des fichiers en mode texte entre différentes plateformes, le codage de fichier par défaut de la plateforme source peut ne pas être pris en charge par la plateforme de destination. Cela entraîne l'échec d'un transfert avec l'erreur suivante:

```
BFGI00058E: The transfer source encoding xxx is illegal or for an unsupported character set.
```

Vous pouvez résoudre cette erreur en définissant le codage source sur un codage pris en charge par la plateforme de destination à l'aide d'une variable d'environnement. Définissez la variable d'environnement système **BFG_JVM_PROPERTIES** sur le système source comme suit: `BFG_JVM_PROPERTIES="-Dfile.encoding=xxx"`, où `xxx` est un codage pris en charge par la plateforme de destination. Par exemple, si vous transférez des fichiers en mode texte depuis une plateforme Solaris vers une autre plateforme et que l'environnement local source est défini sur "ja", définissez **BFG_JVM_PROPERTIES** comme suit: `BFG_JVM_PROPERTIES="-Dfile.encoding=EUC-JP"`. Si l'environnement local source est défini sur "ja_JP.PCK", définissez **BFG_JVM_PROPERTIES** comme suit: `BFG_JVM_PROPERTIES="-Dfile.encoding=Shift_JIS"`.

Vous pouvez également résoudre cette erreur pour un transfert individuel à l'aide du paramètre **-sce** lorsque vous démarrez un nouveau transfert. Pour plus d'informations, voir la section [«fteCreateTransfer: démarrer un nouveau transfert de fichier»](#), à la page 447.

Référence associée

«Propriétés système Java pour MFT», à la page 621

Un certain nombre de propriétés de commande et d'agent Managed File Transfer doivent être définies en tant que propriétés système Java , car elles définissent la configuration de la fonction précoce qui ne peut pas utiliser le mécanisme des propriétés de commande ou d'agent.

Exécution de la trace sous MFT

Vous pouvez utiliser les méthodes décrites dans cette section pour tracer Managed File Transfer.

Pourquoi et quand exécuter cette tâche

Voir [«Traçage des ressources Managed File Transfer sur Multiplatforms»](#), à la page 284 et [«Traçage des ressources Managed File Transfer for z/OS»](#), à la page 291 pour des informations détaillées sur le traçage de différentes ressources Managed File Transfer .

fteSetAgentTraceLevel

La commande **fteSetAgentTraceLevel** permet de modifier le niveau de trace en cours d'un agent de manière dynamique.

Objet

Utilisez cette commande pour activer et désactiver la trace d'agent ou pour modifier le niveau de trace d'agent défini. Lorsque vous utilisez la commande **fteSetAgentTraceLevel** , vous n'avez pas besoin d'arrêter et de redémarrer un agent pour modifier le niveau de trace. Les fichiers de trace générés se trouvent dans `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name/logs/trace%PID%/trace%PID%.txt`, où `%PID%` est l'ID de processus de l'instance d'agent.

**Avertissement :**

Multi Lorsque vous utilisez IBM WebSphere MQ 7.5 ou une version ultérieure sous Multiplateformes, seul l'utilisateur sur lequel s'exécute le processus d'agent peut exécuter la commande **fteSetAgentTraceLevel**.

V 9.0.2 **z/OS** Pour z/OS, pour Continuous Delivery, à partir de IBM MQ 9.0.2, la commande **fteSetAgentTraceLevel** peut être exécutée par:

- Le même ID utilisateur que celui sous lequel le processus d'agent s'exécute.
- Les membres du groupe spécifiés par la propriété d'agent **adminGroup**.

V 9.0.0.1 Cela s'applique également à l'édition Long Term Support de IBM MQ 9.0.0 Fix Pack 1.

Pour plus d'informations, voir la propriété **adminGroup** dans «Le fichier MFT agent.properties», à la page 564.

Dans la IBM WebSphere MQ 7.5 et les versions ultérieures, la commande **fteSetAgentTraceLevel** écrit également une trace pour le contrôleur de processus de l'agent. Les fichiers de trace générés se trouvent dans `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name/logs/pctrace%PID%/pctrace%PID%.txt`, où `%PID%` est l'ID de processus de l'instance d'agent.

Vous pouvez également utiliser la commande pour que le processus d'agent génère un Javacore. L'agent génère un fichier Javacore dans le répertoire suivant: `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name`.

Etant donné que l'exécution de la trace peut affecter vos performances de manière significative et peut produire une grande quantité de données de trace, exécutez la trace avec précaution et uniquement lorsque cela est nécessaire. En règle générale, activez la fonction de trace uniquement lorsque le technicien de maintenance IBM vous le demande.

**Avertissement :**

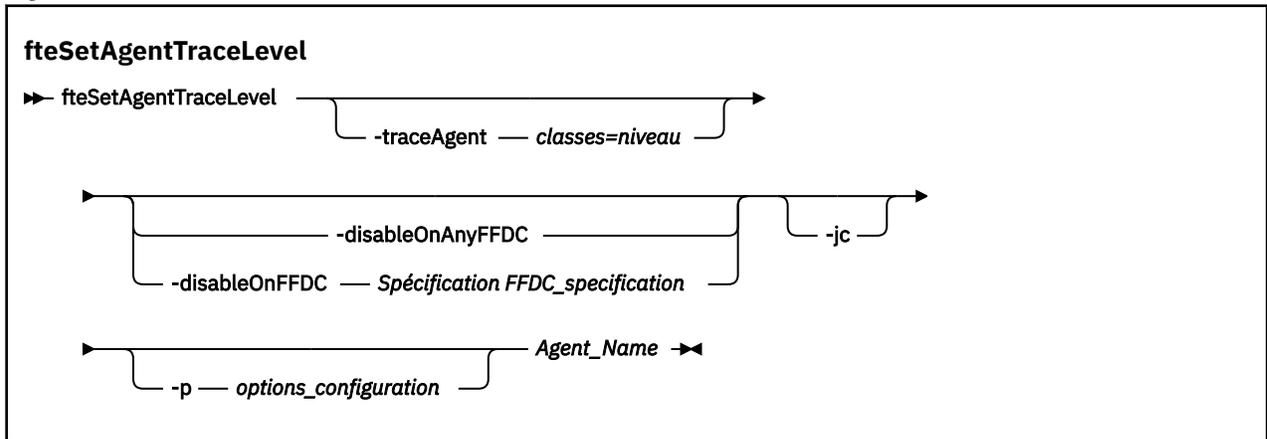
1. Vous devez exécuter cette commande sur le système sur lequel l'agent est en cours d'exécution.
2. Les traces et la consigne ne sont pas conservées lors du redémarrage d'un agent.

Si l'agent s'arrête et est redémarré par le processus du contrôleur de processus, les traces dynamiques et les journaux ne sont pas en vigueur tant que le fichier `agent.properties` n'a pas été mis à jour pour inclure les propriétés de trace et de journal requises.

Vous pouvez définir d'autres propriétés de trace, par exemple la taille du fichier de trace et le nombre de fichiers de trace à conserver, dans le fichier `agent.properties`. Ces propriétés sont décrites dans Propriétés avancées de l'agent.

Spécifiez le paramètre facultatif `-p` pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de votre ensemble par défaut. Pour plus d'informations, voir «Le fichier MFT agent.properties», à la page 564.

Syntaxe



Paramètres

-traceAgent classes=niveau

Obligatoire. Niveau pour définir la trace de l'agent et les classes à appliquer à la trace.

V 9.0.5 Depuis la IBM MQ 9.0.5, vous pouvez spécifier une liste de spécifications de classe séparées par un signe deux-points. Cette option permet de définir le traçage de différentes classes, à différents niveaux. Exemple :

```
fteSetAgentTraceLevel -traceAgent com.ibm.wmqfte.agent=all:com.ibm.wmqfte.cmdhandler=moderate AGENT1
```

Vous pouvez toujours spécifier une liste séparée par des virgules de spécifications de classe auxquelles vous souhaitez que le niveau de trace s'applique. Si vous ne spécifiez pas ce paramètre, le niveau de trace est appliqué à toutes les classes d'agent. Utilisez le format suivant :

```
classes=level
```

Exemple :

```
com.ibm.wmqfte=all
```

Vous pouvez remplacer *classes* par un nom de package Managed File Transfer pour tracer un package spécifique uniquement. Toutefois, comme cette option capture uniquement un sous-ensemble du comportement de l'agent, il n'est généralement pas recommandé d'utiliser le filtrage de package.

Si (*classes*) commencent par un signe plus (+), la liste des classes de trace qui suit le signe plus est ajoutée à toutes les classes de trace existantes actuellement tracées.

Les options de niveau de trace valides sont les suivantes et sont répertoriées par ordre croissant de taille et de détail du fichier de trace:

désactivé

Désactive la trace de l'agent mais continue d'écrire des informations dans les fichiers journaux. Il s'agit de l'option par défaut.

flux

Capture des données pour les points de trace associés au flux de traitement dans l'agent.

modéré

Capture une quantité modérée d'informations de diagnostic dans la trace.

prolix

Capture une quantité prolix d'informations de diagnostic dans la trace.

Tous

Définit la trace d'agent à exécuter sur toutes les classes d'agent.

Pour démarrer le traçage complet de l'agent, exécutez la commande suivante:

```
fteSetAgentTraceLevel -traceAgent =all AGENT_NAME
```

Pour arrêter la fonction de trace complète de l'agent, exécutez la commande suivante:

```
fteSetAgentTraceLevel -traceAgent =off AGENT_NAME
```

-disableOnAnyFFDC

Facultatif. Si ce paramètre est spécifié, la trace est désactivée sur l'agent lorsqu'il génère un fichier FFDC (First Failure Data Capture).

Vous ne pouvez spécifier qu'un seul des paramètres **-disableOnAnyFFDC** et **-disableOnFFDC**.

-disableOnFFDC Spécification FFDC

Facultatif. Si ce paramètre est spécifié, la trace est désactivée sur l'agent lorsqu'il génère un fichier FFDC (First Failure Data Capture) qui correspond à la *spécification FFDC_specification*. *FFDC_specification* est une liste de valeurs séparées par des virgules. Le format des valeurs peut être:

nom_classe

Nom de la classe d'origine de l'outil de diagnostic de premier niveau (FFDC). Par exemple, `com.ibm.wmqfte.classA`.

nom_classe:probe_ID

Nom de la classe et ID sonde de l'emplacement de la classe d'où provient l'outil de diagnostic de premier niveau (FFDC). Par exemple, `com.ibm.wmqfte.classB:1`.

Vous ne pouvez spécifier qu'un seul des paramètres **-disableOnAnyFFDC** et **-disableOnFFDC**.

-jc

Facultatif. Demande à l'agent de générer un fichier javacore. L'équipe du service IBM peut vous demander d'exécuter la commande avec ce paramètre pour faciliter le diagnostic des problèmes. Ce paramètre ne peut être utilisé avec aucun autre paramètre à l'exception de **-p**.

-p options_configuration

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour définir le niveau de trace de l'agent. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

Agent_Name

Obligatoire. Nom du Managed File Transfer Agent pour lequel vous souhaitez définir le niveau de trace.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, le niveau de trace est défini sur `all` pour toutes les classes pour `AGENT1`:

```
fteSetAgentTraceLevel -traceAgent com.ibm.wmqfte=all AGENT1
```

Dans cet exemple, le niveau de trace est défini sur `all` pour les classes `com.ibm.wmqfte.agent.Agent` et `com.ibm.wmqfte.cmdhandler` pour `AGENT1`:

```
fteSetAgentTraceLevel -traceAgent com.ibm.wmqfte.agent.Agent,com.ibm.wmqfte.cmdhandler=moderate AGENT1
```

Dans cet exemple, les sous-classes sont exclues de la trace car le paramètre **-traceLevel** est défini sur `off`. Toutes les classes commençant par `com.ibm.outer` sont tracées au niveau `prolix`, à l'exception des classes commençant par `com.ibm.outer.inner`:

```
fteSetAgentTraceLevel -traceAgent com.ibm.outer=verbose AGENT1  
fteSetAgentTraceLevel -traceAgent +com.ibm.outer.inner=off AGENT1
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Multi *Traçage des ressources Managed File Transfer sur Multiplatforms*

La fonction de trace de Managed File Transfer est fournie pour aider le support IBM à diagnostiquer vos problèmes. Vous pouvez tracer différentes ressources.

Pourquoi et quand exécuter cette tâche

Consultez

- «[Traçage des agents Managed File Transfer sur Multiplatforms](#)», à la [page 284](#) pour plus d'informations sur la façon dont vous tracez les agents.
- «[Traçage des commandes Managed File Transfer sur Multiplatforms](#)», à la [page 287](#) pour plus d'informations sur la façon dont vous tracez les commandes.

Multi *Traçage des agents Managed File Transfer sur Multiplatforms*

La fonction de trace de Managed File Transfer est fournie pour aider le support IBM à diagnostiquer vos problèmes. Diverses commandes et propriétés contrôlent le comportement de cette fonction.

Pourquoi et quand exécuter cette tâche

Si vous êtes invité à fournir une sortie de trace pour examiner un problème lié à un agent, utilisez l'une des options suivantes, selon qu'il est possible d'arrêter l'agent pendant une courte période.

Si vous ne savez pas quelle option utiliser, contactez votre représentant du support IBM qui vous conseillera sur la meilleure façon de collecter des données de trace pour le problème que vous rencontrez.

Procédure

- S'il est possible d'arrêter un agent pendant une courte période, collectez une trace de l'agent au démarrage.
Pour plus d'informations, voir «[Collecte d'une trace d'agent Managed File Transfer au démarrage](#)», à la [page 285](#).
- S'il n'est pas possible d'arrêter un agent, collectez une trace de manière dynamique à l'aide de la commande **fteSetAgentTraceLevel**.
Pour plus d'informations, voir «[Collecte dynamique d'une trace d'agent Managed File Transfer](#)», à la [page 286](#).

Lorsqu'il est possible d'arrêter un agent pendant une courte période, vous devez collecter la trace de l'agent Managed File Transfer depuis le démarrage.

Avant de commencer

Vous devez définir différentes propriétés dans le fichier `agent.properties` pour l'agent à tracer.

Pourquoi et quand exécuter cette tâche

Pour collecter une trace à partir du démarrage, procédez comme suit.

Procédure

1. Localisez le fichier `agent.properties` de l'agent qui doit être tracé.

Le fichier `agent.properties` se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name`.

2. Editez les fichiers et ajoutez des entrées pour les propriétés suivantes:

- **trace**=*spécification de trace*

La propriété **trace** détermine les classes internes et les packages à tracer. Sauf indication contraire de votre représentant du support IBM, définissez cette propriété sur la valeur `com.ibm.wmqfte=all`.

- **traceFiles**=*nombre de fichiers de trace à utiliser*

- **traceSize**=*taille de chaque fichier de trace, en Mo*

Les propriétés **traceFiles** et **traceSize** sont utilisées pour contrôler la quantité de données de trace collectées. Vous devez définir ces propriétés sur des valeurs importantes afin de collecter autant de données de trace que possible.

Par exemple, pour collecter 1GB de trace d'encapsulation à l'aide de la spécification de trace `com.ibm.wmqfte=all`, ajoutez les lignes suivantes au fichier `agent.properties` :

```
trace=com.ibm.wmqfte=all
traceFiles=5
traceSize=200
```

Par conséquent, l'agent écrit les données de trace dans un maximum de 5 fichiers, chaque fichier ayant une taille de 200MB.

Pour plus d'informations sur ces propriétés d'agent, voir [Le fichier MFT agent.properties](#).

3. Arrêtez l'agent à tracer à l'aide de la commande **fteStopAgent**.
4. Démarrez l'agent en exécutant la commande **fteStartAgent**.
5. Reproduisez le problème.
6. Arrêtez l'agent.
7. Editez le fichier `agent.properties` de l'agent et supprimez les entrées des propriétés **trace**, **traceFiles** et **traceSize** que vous avez ajoutées à l'étape «2», à la page 285.

Cela permet de s'assurer que la trace n'est pas activée lors du prochain redémarrage de l'agent.

Résultats

Les fichiers de trace résultants sont écrits dans le répertoire `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name/logs/trace%PID%`, où `%PID%` est l'identificateur de processus de l'agent.

La commande **fteSetAgentTraceLevel** permet de collecter la trace à partir d'un agent en cours d'exécution. Cela peut s'avérer très utile si le support IBM a besoin de voir une trace provenant d'un agent qui ne peut pas être arrêté.

Pourquoi et quand exécuter cette tâche

Pour collecter une trace à partir d'un agent à l'aide de la commande **fteSetAgentTraceLevel**, procédez comme suit.

Procédure

1. Activez la trace pour l'agent en exécutant la commande suivante:

```
fteSetAgentTraceLevel -traceAgent classes=level agent_name
```

Le paramètre `-traceAgent` détermine les classes internes et les packages à tracer. Sauf indication contraire de votre représentant du support IBM, définissez cette propriété sur la valeur `com.ibm.wmqfte=all`.

2. Reproduisez le problème.
3. Désactivez la trace pour l'agent en exécutant la commande suivante:

```
fteSetAgentTraceLevel -traceAgent =off agent_name
```

Si un agent est occupé, les fichiers de trace peuvent s'encapsuler rapidement et écraser les informations nécessaires pour identifier le problème. Si tel est le cas, planifiez un certain temps pour arrêter l'agent, puis procédez comme indiqué dans les étapes suivantes. Si vous ne pouvez pas arrêter l'agent pendant une courte période, contactez votre représentant du support IBM et discutez des autres spécifications de trace à utiliser afin de réduire la quantité de données de trace générées.

4. Localisez le fichier `agent.properties` de l'agent qui doit être tracé.

Le fichier `agent.properties` se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name`.

5. Editez le fichier et ajoutez des entrées pour les propriétés suivantes:

```
traceFiles=number_of_trace_files_to_use  
traceSize=size_of_each_trace_file_in_MB
```

Les propriétés **traceFiles** et **traceSize** sont utilisées pour contrôler la quantité de données de trace collectées.

La valeur par défaut de la propriété **traceFiles** est 5 et celle de la propriété **traceSize** est 20MB. Cela signifie que si vous activez la trace de manière dynamique et que vous n'avez pas défini les propriétés, l'agent écrit les informations de trace dans 5 fichiers de trace d'encapsulation, chacun d'eux ayant une taille maximale de 20MB.

Vous devez définir ces propriétés sur des valeurs importantes afin de collecter autant de données de trace que possible.

Par exemple, pour collecter 1GB de trace d'encapsulation, ajoutez les lignes suivantes au fichier `agent.properties`:

```
traceFiles=5  
traceSize=200
```

Par conséquent, l'agent écrit les données de trace dans un maximum de 5 fichiers, chaque fichier ayant une taille de 200MB.

Pour plus d'informations sur ces propriétés d'agent, voir [Le fichier MFT agent.properties](#).

6. Arrêtez l'agent en exécutant la commande **fteStopAgent**.

7. Démarrez l'agent en exécutant la commande **fteStartAgent** .
8. Activez la trace pour l'agent en exécutant la commande suivante:

```
fteSetAgentTraceLevel -traceAgent classes=level agent_name
```

Sauf indication contraire de votre représentant du support IBM , définissez la propriété **-traceAgent** sur la valeur `com.ibm.wmqfte=all`.

9. Reproduisez le problème.
10. Désactivez la fonction de trace sur l'agent en exécutant la commande suivante:

```
fteSetAgentTraceLevel -traceAgent =off agent_name
```

Résultats

Les fichiers de trace résultants sont écrits dans le répertoire `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name/logs/trace%PID%`, où `%PID%` est l'identificateur de processus de l'agent.

Multi *Traçage des commandes Managed File Transfer sur Multiplatforms*

La fonction de trace de Managed File Transfer est fournie pour aider le support IBM à diagnostiquer vos problèmes. Vous pouvez utiliser cette fonction pour tracer des commandes.

Pourquoi et quand exécuter cette tâche



Avvertissement : Le traçage d'une commande collecte uniquement des informations sur le traitement effectué par la commande. Il ne trace aucune activité qu'un agent peut effectuer lors du traitement de cette commande.

Procédure

1. Ouvrez une invite de commande et accédez au répertoire `MQ_INSTALLATION_PATH\bin` .
2. Exécutez la commande :

Linux UNIX

```
./command_name -trace classes=level -tracePath directory_path command_arguments
```

Windows

```
command_name -trace classes=level -tracePath directory_path command_arguments
```

Où

- `nom_commande` est le nom de la commande à tracer.
- `classes=level` est le niveau de trace à utiliser et les classes pour lesquelles activer la trace. Sauf indication contraire de votre représentant de support IBM , définissez ce paramètre sur `com.ibm.wmqfte=all`.
- `directory_path` est le répertoire dans lequel les fichiers de trace seront écrits.
- `command_arguments` sont les arguments qui doivent être transmis à la commande, par exemple, le nom de l'agent pour la commande **ftePingAgent** .

Résultats

Les fichiers de trace résultants sont écrits dans le répertoire spécifié par le paramètre **-tracePath** .

Les fichiers de trace sont appelés `trace%PID%.txt.number`, où:

- `%PID%` est l'identificateur de processus de la commande.

- *numéro* est un numéro de séquence du fichier de trace. En règle générale, les informations de trace générées par une commande sont contenues dans un fichier de trace unique dont le numéro de séquence est 0.

Toutefois, il est possible qu'une commande génère beaucoup d'informations de trace. Dans ce cas, la trace sera écrite dans plusieurs fichiers. Le fichier de trace en cours a le numéro de séquence 0, le fichier de trace le plus ancien suivant a le numéro de séquence 1, etc.

La sortie de trace des commandes est écrite avec un maximum de cinq fichiers de trace en boucle. La taille maximale de chaque fichier de trace est de 20MB.

Remarque : Si l'utilisateur exécutant la commande n'est pas autorisé à écrire dans le répertoire spécifié par le paramètre **-tracePath**, la sortie de trace est écrite dans l'erreur standard.

Exemple

Dans cet exemple, la commande **fteListAgents** est tracée et la trace est écrite dans le répertoire C:\trace :

```
fteListAgents -trace com.ibm.wmqfte=all -tracePath C:\trace
```

Dans cet exemple, la commande **fteCreateTransfer** est tracée et la trace est écrite dans le répertoire /tmp :

```
fteCreateTransfer -trace com.ibm.wmqfte=all -tracePath /tmp -t text -sa AGENT1
-da AGENT2 -df /import/transferredfile.txt /export/originalfile.txt
```

Le fichier de trace écrit dans /tmp contient uniquement des informations sur le traitement effectué par la commande **fteCreateTransfer**, telles que la façon dont la commande génère le message de demande de transfert envoyé à l'agent et la durée pendant laquelle elle attend que l'agent renvoie un accusé de réception indiquant qu'il a reçu la demande. Le fichier de trace ne contient aucune information sur le transfert lui-même.

Multi *Traçage des consignateurs autonomes Managed File Transfer sur Multiplatforms*

La fonction de trace de Managed File Transfer est fournie pour aider le support IBM à diagnostiquer vos problèmes. Diverses commandes et propriétés contrôlent le comportement de cette fonction.

Pourquoi et quand exécuter cette tâche

Si vous êtes invité à fournir une sortie de trace pour examiner un problème lié à un consignateur, utilisez l'une des options suivantes, selon qu'il est possible d'arrêter le consignateur pendant une courte période.

Si vous ne savez pas quelle option utiliser, contactez votre représentant du support IBM qui vous conseillera sur la meilleure façon de collecter des données de trace pour le problème que vous rencontrez.

Procédure

- S'il est possible d'arrêter un consignateur pendant une courte période, collectez une trace du consignateur à partir du démarrage.
Voir [«Collecte d'une trace de consignateur autonome Managed File Transfer depuis le démarrage»](#), à la page 288.
- S'il n'est pas possible d'arrêter un consignateur, collectez une trace de manière dynamique à l'aide de la commande **fteSetLoggerTraceLevel**.
Voir [«Collecte dynamique d'une trace de consignateur autonome Managed File Transfer»](#), à la page 289.

Multi *Collecte d'une trace de consignateur autonome Managed File Transfer depuis le démarrage*

Lorsqu'il est possible d'arrêter un consignateur pendant une courte période, vous devez collecter la trace du consignateur Managed File Transfer à partir du démarrage.

Avant de commencer

Vous devez définir différentes propriétés dans le fichier `logger.properties` pour le consignateur qui doit être tracé.

Pourquoi et quand exécuter cette tâche

Pour collecter une trace à partir du démarrage, procédez comme suit.

Procédure

1. Localisez le fichier `logger.properties` du consignateur qui doit être tracé.

Le fichier `logger.properties` se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

2. Editez le fichier et ajoutez des entrées pour les propriétés suivantes:

- **trace**=*spécification de trace*

La propriété **trace** détermine les classes internes et les packages à tracer. Sauf indication contraire de votre représentant du support IBM, définissez cette propriété sur la valeur `com.ibm.wmqfte=all`.

- **traceFiles**=*nombre de fichiers de trace à utiliser*
- **traceSize**=*taille de chaque fichier de trace, en Mo*

Les propriétés **traceFiles** et **traceSize** sont utilisées pour contrôler la quantité de données de trace collectées. Vous devez définir ces propriétés sur des valeurs importantes afin de collecter autant de données de trace que possible.

Par exemple, pour collecter 1GB de trace d'encapsulation à l'aide de la spécification de trace `com.ibm.wmqfte=all`, ajoutez les lignes suivantes au fichier `logger.properties`:

```
trace=com.ibm.wmqfte=all
traceFiles=5
traceSize=200
```

Ainsi, le consignateur écrit les données de trace dans un maximum de 5 fichiers, chaque fichier ayant une taille de 200MB.

Pour plus d'informations sur ces propriétés de consignateur, voir [Le fichier MFT logger.properties](#).

3. Arrêtez le consignateur à tracer à l'aide de la commande **fteStopLogger**.
4. Démarrez le consignateur en exécutant la commande **fteStartLogger**.
5. Reproduisez le problème.
6. Arrêtez le consignateur.
7. Editez le fichier `logger.properties` du consignateur et supprimez les entrées des propriétés **trace**, **traceFiles** et **traceSize** que vous avez ajoutées à l'étape «2», à la page 289.
Cela permet de s'assurer que la trace n'est pas activée lors du prochain redémarrage du consignateur.

Résultats

Les fichiers de trace résultants sont écrits dans le répertoire `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/loggers/logger_name/logs/trace%PID%`, où `%PID%` est l'identificateur de processus du consignateur.

Multi Collecte dynamique d'une trace de consignateur autonome Managed File Transfer

La commande **fteSetLoggerTraceLevel** permet de collecter la trace à partir d'un consignateur en cours d'exécution. Cela peut être très utile si le support IBM a besoin de voir une trace d'un consignateur qui ne peut pas être arrêtée.

Pourquoi et quand exécuter cette tâche

Pour collecter une trace à partir d'un consigneur à l'aide de la commande **fteSetLoggerTraceLevel** , procédez comme suit.

Procédure

1. Activez la trace pour le consigneur en exécutant la commande suivante:

```
fteSetLoggerTraceLevel -traceLogger classes=level logger_name
```

Le paramètre `-traceLogger` détermine les classes internes et les packages à tracer. Sauf indication contraire de votre représentant du support IBM , définissez cette propriété sur la valeur `com.ibm.wmqfte=all`.

2. Reproduisez le problème.
3. Désactivez la fonction de trace à l'aide de la commande suivante:

```
fteSetLoggerTraceLevel -traceLogger =off logger_name
```

4. Si un consigneur est occupé, il est possible que les fichiers de trace soient encapsulés rapidement et écrasent les informations nécessaires pour identifier le problème.

Si vous pouvez arrêter le consigneur pendant une courte période, procédez comme suit pour réduire la quantité de données de trace collectées. Sinon, contactez le support IBM et discutez des autres spécifications de trace pour réduire la quantité de données de trace collectées.

- a) Planifiez un certain temps pour arrêter le consigneur.
- b) Localisez le fichier `logger.properties` du consigneur qui doit être tracé.

Le fichier `logger.properties` se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

- c) Editez le fichier et ajoutez des entrées pour les propriétés suivantes:

```
traceFiles=nombre_fichiers_trace_à_utiliser  
traceSize=taille_du_fichier_trace_en_Mo
```

Les propriétés **traceFiles** et **traceSize** sont utilisées pour contrôler la quantité de données de trace collectées.

La valeur par défaut de la propriété **traceFiles** est 5 et celle de la propriété **traceSize** est 20MB. Cela signifie que si vous activez la trace de manière dynamique et que vous n'avez pas défini les propriétés, l'agent écrit les informations de trace dans 5 fichiers de trace d'encapsulation, chacun d'eux ayant une taille maximale de 20MB.

Vous devez définir ces propriétés sur des valeurs importantes afin de collecter autant de données de trace que possible.

Par exemple, pour collecter 1GB de trace d'encapsulation, ajoutez les lignes suivantes au fichier `logger.properties` :

```
traceFiles=5  
traceSize=200
```

Ainsi, le consigneur écrit les données de trace dans un maximum de 5 fichiers, chaque fichier ayant une taille de 200MB.

Pour plus d'informations sur ces propriétés de consigneur, voir [Le fichier MFT logger.properties](#).

- d) Arrêtez le consigneur en exécutant la commande **fteStopLogger** .
- e) Démarrez le consigneur en exécutant la commande **fteStartLogger** .

- f) Activez la fonction de trace pour le consignateur en exécutant la commande suivante. Sauf indication contraire de votre représentant du support IBM, définissez la propriété **-traceLogger** sur la valeur `com.ibm.wmqfte=all`.

```
fteSetLoggerTraceLevel -traceLogger classes=level logger_name
```

- g) Reproduisez le problème.
h) Désactivez la trace pour le consignateur en exécutant la commande suivante:

```
fteSetLoggerTraceLevel -traceLogger =off logger_name
```

Résultats

Les fichiers de trace résultants sont écrits dans le répertoire `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/loggers/logger_name/logs/trace%PID%`, où %PID% est l'identificateur de processus du consignateur.

Traçage des ressources Managed File Transfer for z/OS

La fonction de trace de Managed File Transfer for z/OS est fournie pour aider le support IBM à diagnostiquer vos problèmes. Vous pouvez tracer différentes ressources.

Pourquoi et quand exécuter cette tâche

Consultez

- «Traçage des agents Managed File Transfer for z/OS», à la page 291 pour plus d'informations sur la façon dont vous tracez les agents.
- «Commandes Managed File Transfer for z/OS de traçage», à la page 297 pour plus d'informations sur la façon dont vous tracez les commandes.

Traçage des agents Managed File Transfer for z/OS

La fonction de trace de Managed File Transfer for z/OS est fournie pour aider le support IBM à diagnostiquer vos problèmes. Diverses commandes et propriétés contrôlent le comportement de cette fonction.

Pourquoi et quand exécuter cette tâche

Si vous êtes invité à fournir une sortie de trace pour examiner un problème avec un agent, utilisez l'une des options suivantes.

Si vous ne savez pas quelle option utiliser, contactez votre représentant du support IBM qui vous conseillera sur la meilleure façon de collecter des données de trace pour le problème que vous rencontrez.

Procédure

- S'il est possible d'arrêter un agent pendant une courte période, collectez une trace de l'agent au démarrage.
Pour plus d'informations, voir [«Collecte d'une trace d'agent Managed File Transfer for z/OS au démarrage»](#), à la page 292.
- S'il n'est pas possible d'arrêter un agent, collectez une trace de manière dynamique à l'aide de la commande **fteSetAgentTraceLevel**.
Pour plus d'informations, voir [«Collecte dynamique d'une trace d'agent Managed File Transfer for z/OS»](#), à la page 294.

z/OS Collecte d'une trace d'agent Managed File Transfer for z/OS au démarrage

Lorsqu'il est possible d'arrêter un agent pendant une courte période, vous devez collecter la trace de l'agent IBM MQ Managed File Transfer à partir du démarrage.

Pourquoi et quand exécuter cette tâche

Le mode de collecte de la trace varie selon que l'agent est administré à l'aide des services USS (Unix System Services) ou JCL.

Si vous ne savez pas laquelle des options suivantes utiliser, contactez votre représentant du support IBM qui vous conseillera sur la meilleure façon de collecter des données de trace pour le problème que vous rencontrez.

Procédure

- Si vous utilisez z/OS UNIX, voir [«Collecte d'une trace d'agent à partir du démarrage à l'aide d'USS»](#), à la page 292.
- Si vous utilisez JCL, voir [«Collecte d'une trace d'agent à partir du démarrage à l'aide de JCL»](#), à la page 293.

z/OS Collecte d'une trace d'agent à partir du démarrage à l'aide d'USS

Pour collecter une trace d'un agent Managed File Transfer for z/OS administré à l'aide des services système UNIX (USS) à partir du démarrage, vous devez définir les différentes propriétés requises dans le fichier `agent.properties` pour cet agent avant de le démarrer.

Pourquoi et quand exécuter cette tâche

Pour collecter une trace à partir du démarrage, procédez comme suit.

Procédure

1. Localisez le fichier `agent.properties` de l'agent qui doit être tracé.

Le fichier `agent.properties` se trouve dans le répertoire `BFG_DATA/mqft/config/coordination_qmgr_name/agents/agent_name`.

2. Editez les fichiers et ajoutez des entrées pour les propriétés suivantes:

- **trace**=spécification de trace

La propriété **trace** détermine les classes internes et les packages à tracer. Sauf indication contraire de votre représentant du support IBM, définissez cette propriété sur la valeur `com.ibm.wmqfte=all`.

- **traceFiles**=nombre de fichiers de trace à utiliser
- **traceSize**=taille de chaque fichier de trace, en Mo

Les propriétés **traceFiles** et **traceSize** sont utilisées pour contrôler la quantité de données de trace collectées. Vous devez définir ces propriétés sur des valeurs importantes afin de collecter autant de données de trace que possible.

Par exemple, pour collecter 1GB de trace d'encapsulation à l'aide de la spécification de trace `com.ibm.wmqfte=all`, ajoutez les lignes suivantes au fichier `agent.properties` :

```
trace=com.ibm.wmqfte=all
traceFiles=5
traceSize=200
```

Par conséquent, l'agent écrit les données de trace dans un maximum de 5 fichiers, chaque fichier ayant une taille de 200MB.

Pour plus d'informations sur ces propriétés d'agent, voir [Le fichier MFT agent.properties](#).

3. Arrêtez l'agent à tracer à l'aide de la commande **fteStopAgent** .
4. Démarrez l'agent en exécutant la commande **fteStartAgent** .
5. Reproduisez le problème.
6. Arrêtez l'agent.
7. Editez le fichier `agent.properties` de l'agent et supprimez les entrées des propriétés **trace**, **traceFiles** et **traceSize** que vous avez ajoutées à l'étape «2», à la page 292.
Cela permet de s'assurer que la trace n'est pas activée lors du prochain redémarrage de l'agent.

Résultats

Les fichiers de trace résultants sont écrits dans le répertoire `BFG_DATA/mqft/logs/coordination_qmgr_name/agents/agent_name/logs/trace%PID%`, où `%PID%` est l'identificateur de processus de l'agent.

Collecte d'une trace d'agent à partir du démarrage à l'aide de JCL

Pour collecter une trace d'un agent Managed File Transfer for z/OS administré à l'aide de JCL depuis le démarrage, vous devez définir diverses propriétés requises dans le fichier `agent.properties` de cet agent avant de le démarrer.

Pourquoi et quand exécuter cette tâche

Pour collecter une trace à partir du démarrage, procédez comme suit.

Procédure

1. Localisez le fichier `agent.properties` de l'agent qui doit être tracé.
Le fichier `agent.properties` se trouve dans le répertoire `BFG_DATA/mqft/config/coordination_qmgr_name/agents/agent_name` .

2. Editez les fichiers et ajoutez des entrées pour les propriétés suivantes:

- **trace**=*spécification de trace*

La propriété **trace** détermine les classes internes et les packages à tracer. Sauf indication contraire de votre représentant du support IBM , définissez cette propriété sur la valeur `com.ibm.wmqfte=all`.

- **traceFiles**=*nombre de fichiers de trace à utiliser*
- **traceSize**=*taille de chaque fichier de trace, en Mo*

Les propriétés **traceFiles** et **traceSize** sont utilisées pour contrôler la quantité de données de trace collectées. Vous devez définir ces propriétés sur des valeurs importantes afin de collecter autant de données de trace que possible.

Par exemple, pour collecter 1GB de trace d'encapsulation à l'aide de la spécification de trace `com.ibm.wmqfte=all`, ajoutez les lignes suivantes au fichier `agent.properties` :

```
trace=com.ibm.wmqfte=all
traceFiles=5
traceSize=200
```

Par conséquent, l'agent écrit les données de trace dans un maximum de 5 fichiers, chaque fichier ayant une taille de 200MB.

Pour plus d'informations sur ces propriétés d'agent, voir [Le fichier MFT agent.properties](#).

3. Localisez le fichier contenant le JCL de l'agent qui doit être tracé.
4. Soumettez le membre BFGAGSP dans le fichier pour arrêter l'agent.
5. Redémarrez l'agent en soumettant le membre BFGAGST dans le fichier.
6. Reproduisez le problème.
7. Soumettez le membre BFGAGSP dans le fichier pour arrêter à nouveau l'agent.

8. Editez le fichier `agent.properties` de l'agent et supprimez les entrées des propriétés **trace**, **traceFiles** et **traceSize** que vous avez ajoutées à l'étape «2», à la page 293.

Cela permet de s'assurer que la trace n'est pas activée lors du prochain redémarrage de l'agent.

Résultats

Les fichiers de trace résultants sont écrits dans le répertoire `BFG_DATA/mqft/logs/coordination_qmgr_name/agents/agent_name/logs/trace%PID%`, où `%PID%` est l'identificateur de processus de l'agent.

Collecte dynamique d'une trace d'agent Managed File Transfer for z/OS

Lorsqu'il n'est pas possible d'arrêter un agent pendant une courte période, vous devez collecter la trace de l'agent Managed File Transfer for z/OS de manière dynamique.

Pourquoi et quand exécuter cette tâche

Le mode de collecte de la trace varie selon que l'agent est administré à l'aide des services USS (Unix System Services) ou JCL.

Si vous ne savez pas laquelle des options suivantes utiliser, contactez votre représentant du support IBM qui vous conseillera sur la meilleure façon de collecter des données de trace pour le problème que vous rencontrez.

Procédure

- Si vous utilisez:
 - Unix System Services (USS), voir [«Collecte dynamique d'une trace d'agent à l'aide d'USS»](#), à la page 294.
 - JCL, voir [«Collecte dynamique d'une trace d'agent à l'aide de JCL»](#), à la page 295.

Collecte dynamique d'une trace d'agent à l'aide d'USS

La commande **fteSetAgentTraceLevel** permet de collecter la trace à partir d'un agent en cours d'exécution. Cela peut s'avérer très utile si le support IBM a besoin de voir une trace provenant d'un agent qui ne peut pas être arrêté.

Pourquoi et quand exécuter cette tâche

Pour collecter une trace à partir d'un agent Managed File Transfer for z/OS à l'aide de la commande **fteSetAgentTraceLevel**, procédez comme suit.

Procédure

1. Activez la fonction de trace pour l'agent en exécutant la commande suivante:

```
fteSetAgentTraceLevel -traceAgent classes=level agent_name
```

Le paramètre `-traceAgent` détermine les classes internes et les packages à tracer. Sauf indication contraire de votre représentant du support IBM, définissez cette propriété sur la valeur `com.ibm.wmqfte=all`.

2. Reproduisez le problème.
3. Désactivez la fonction de trace pour l'agent en exécutant la commande suivante:

```
fteSetAgentTraceLevel -traceAgent =off agent_name
```

4. Si un agent est occupé, les fichiers de trace peuvent s'encapsuler rapidement et écraser les informations nécessaires pour identifier le problème.

Si vous pouvez arrêter l'agent pendant une courte période, procédez comme suit pour réduire la quantité de données de trace collectées. Sinon, contactez le support IBM et discutez des autres spécifications de trace pour réduire la quantité de données de trace collectées.

- a) Planifiez un certain temps pour arrêter l'agent.
- b) Localisez le fichier `agent.properties` de l'agent qui doit être tracé.

Le fichier `agent.properties` se trouve dans le répertoire `BFG_DATA/mqft/config/coordination_qmgr_name/agents/agent_name`.

- c) Editez le fichier et ajoutez des entrées pour les propriétés suivantes:

traceFiles=nombre_fichiers_trace_à_utiliser
traceSize=taille_du_fichier_trace_en_Mo

Les propriétés **traceFiles** et **traceSize** sont utilisées pour contrôler la quantité de données de trace collectées.

La valeur par défaut de la propriété **traceFiles** est 5 et celle de la propriété **traceSize** est 20MB. Cela signifie que si vous activez la trace de manière dynamique et que vous n'avez pas défini les propriétés, l'agent écrit les informations de trace dans 5 fichiers de trace d'encapsulation, chacun d'eux ayant une taille maximale de 20MB.

Vous devez définir ces propriétés sur des valeurs importantes afin de collecter autant de données de trace que possible.

Par exemple, pour collecter 1GB de trace d'encapsulation, ajoutez les lignes suivantes au fichier `agent.properties` :

```
traceFiles=5
traceSize=200
```

Par conséquent, l'agent écrit les données de trace dans un maximum de 5 fichiers, chaque fichier ayant une taille de 200MB.

Pour plus d'informations sur ces propriétés d'agent, voir [Le fichier MFT agent.properties](#).

- d) Arrêtez l'agent en exécutant la commande **fteStopAgent**.
- e) Démarrez l'agent en exécutant la commande **fteStartAgent**.
- f) Activez la fonction de trace pour l'agent en exécutant la commande suivante:

```
fteSetAgentTraceLevel -traceAgent trace_specification agent_name
```

- g) Reproduisez le problème.
- h) Désactivez la fonction de trace pour l'agent en exécutant la commande suivante:

```
fteSetAgentTraceLevel -traceAgent =off agent_name
```

Résultats

Les fichiers de trace résultants sont écrits dans le répertoire `BFG_DATA/mqft/logs/coordination_qmgr_name/agents/agent_name/logs/trace%PID%`, où `%PID%` est l'identificateur de processus de l'agent.

Collecte dynamique d'une trace d'agent à l'aide de JCL

Vous pouvez utiliser le membre `BFGAGTC` dans le fichier contenant le JCL, pour l'agent à tracer, afin de collecter la trace à partir d'un agent Managed File Transfer for z/OS en cours d'exécution. Cela peut s'avérer très utile si le support IBM a besoin d'afficher une trace à partir d'un agent qui ne peut pas être arrêté.

Pourquoi et quand exécuter cette tâche

Pour collecter une trace à partir d'un agent à l'aide du membre BFGAGTC, procédez comme suit.

Procédure

1. Localisez le fichier contenant le JCL de l'agent qui doit être tracé.
2. Editez le membre BFGAGTC dans le fichier et localisez la ligne qui contient le texte:

```
-traceAgent
```

Le texte qui suit contient la liste des classes internes et des packages à tracer. Par défaut, cette liste est définie sur:

```
com.ibm.wmqfte=all
```

Sauf indication contraire de votre représentant du support IBM, conservez cette valeur telle qu'elle est.

3. Soumettez le membre BFGAGTC.
4. Reproduisez le problème.
5. Editez à nouveau le membre BFGAGTC et définissez le paramètre **-traceAgent** sur = *off*, comme suit:

```
-traceAgent =off +
```

6. Soumettez à nouveau le membre BFGAGTC pour désactiver la trace.
7. Si un agent est occupé, il est possible que les fichiers de trace soient rapidement encapsulés et écrasent les informations nécessaires pour identifier le problème.

Dans cette situation, il existe deux options:

a) La première option consiste à:

- i) Planifiez un certain temps pour arrêter l'agent.
- ii) Localisez le fichier `agent.properties` de l'agent qui doit être tracé. Le fichier `agent.properties` se trouve dans le répertoire `BFG_DATA/mqft/config/coordination_qmgr_name/agents/agent_name`.
- iii) Editez les fichiers et ajoutez des entrées pour les propriétés suivantes:

- **traceFiles**=nombre de fichiers de trace à utiliser
- **traceSize**=taille de chaque fichier de trace, en Mo

Les propriétés **traceFiles** et **traceSize** sont utilisées pour contrôler la quantité de données de trace collectées.

La valeur par défaut de la propriété **traceFiles** est 5 et celle de la propriété **traceSize** est 20MB. Cela signifie que si vous activez la trace de manière dynamique et que vous n'avez pas défini les propriétés, l'agent écrit les informations de trace dans 5 fichiers de trace d'encapsulation, chacun d'eux ayant une taille maximale de 20MB.

Vous devez définir ces propriétés sur des valeurs importantes afin de collecter autant de données de trace que possible.

Par exemple, pour collecter 1GB de trace d'encapsulation, ajoutez les lignes suivantes au fichier `agent.properties` :

```
traceFiles=5  
traceSize=200
```

Par conséquent, l'agent écrit les données de trace dans un maximum de 5 fichiers, chaque fichier ayant une taille de 200MB.

Pour plus d'informations sur ces propriétés d'agent, voir [Le fichier MFT agent.properties](#).

- iv) Localisez le fichier contenant le JCL de l'agent qui doit être tracé.
- v) Soumettez le membre BFGAGSP dans le fichier pour arrêter l'agent.
- vi) Redémarrez l'agent en soumettant le membre BFGAGST dans le fichier.
- vii) Editez le membre BFGAGTC dans le fichier et localisez la ligne qui contient le texte:

```
-traceAgent
```

Le texte qui suit contient la liste des classes internes et des packages à tracer. Par défaut, cette liste est définie sur:

```
com.ibm.wmqfte=all
```

Sauf indication contraire de votre représentant du support IBM, conservez cette valeur telle qu'elle est.

- viii) Lorsqu'il est temps d'activer la trace, soumettez le membre BFGAGTC.
- ix) Reproduisez le problème.
- x) Editez à nouveau le membre BFGAGTC et définissez le paramètre **-traceAgent** sur = *off*, comme suit:

```
-traceAgent =off +
```

- xi) Soumettez à nouveau le membre BFGAGTC pour désactiver la trace.
- b) La deuxième option consiste à contacter votre représentant de support IBM s'il n'est pas possible d'arrêter l'agent pendant une courte période.
- Vous pouvez ensuite discuter d'autres spécifications de trace à utiliser afin de réduire la quantité de données de trace générées.

Résultats

Les fichiers de trace résultants sont écrits dans le répertoire *BFG_DATA/mqft/logs/coordination_qmgr_name/agents/agent_name/logs/trace%PID%*, où %PID% est l'identificateur de processus de l'agent.

Commandes Managed File Transfer for z/OS de traçage

La fonction de trace de Managed File Transfer for z/OS est fournie pour aider le support IBM à diagnostiquer vos problèmes. Vous pouvez utiliser cette fonction pour tracer des commandes.

Pourquoi et quand exécuter cette tâche



Avertissement : Le traçage d'une commande collecte uniquement des informations sur le traitement effectué par la commande. Il ne trace aucune activité qu'un agent peut effectuer lors du traitement de cette commande.

Le mode de collecte de la trace varie selon que la commande est exécutée à l'aide des services USS (Unix System Services) ou JCL.

Si vous ne savez pas quelle option utiliser, contactez votre représentant du support IBM qui vous conseillera sur la meilleure façon de collecter des données de trace pour le problème que vous rencontrez.

Procédure

- Si vous utilisez z/OS UNIX, voir [«Collecte d'une trace d'agent à partir du démarrage à l'aide d'USS»](#), à la page 292.
- Si vous utilisez JCL, voir [«Collecte d'une trace d'agent à partir du démarrage à l'aide de JCL»](#), à la page 293.

Pour collecter une trace d'une commande Managed File Transfer for z/OS à l'aide des services système UNIX (USS), procédez comme suit.

Procédure

1. Ouvrez une invite de commande et accédez au répertoire *BFG_PROD/bin*.
2. Exécutez la commande :

```
./command_name -trace classes=level -tracePath directory_path command_arguments
```

Où

- *nom_commande* est le nom de la commande à tracer.
- *classes=level* est le niveau de trace à utiliser et les classes pour lesquelles activer la trace. Sauf indication contraire de votre représentant de support IBM, définissez ce paramètre sur `com.ibm.wmqfte=all`.
- *directory_path* est le répertoire dans lequel les fichiers de trace seront écrits.
- *command_arguments* sont les arguments qui doivent être transmis à la commande, par exemple, le nom de l'agent pour la commande **ftePingAgent**.

Résultats

Les fichiers de trace résultants sont écrits dans le répertoire spécifié par le paramètre **-tracePath**.

Les fichiers de trace sont appelés `trace%PID%.txt.number`, où:

- `%PID%` est l'identificateur de processus de la commande.
- *numéro* est un numéro de séquence du fichier de trace. En règle générale, les informations de trace générées par une commande sont contenues dans un fichier de trace unique dont le numéro de séquence est 0.

Toutefois, il est possible qu'une commande génère beaucoup d'informations de trace. Dans ce cas, la trace sera écrite dans plusieurs fichiers. Le fichier de trace en cours a le numéro de séquence 0, le fichier de trace le plus ancien suivant a le numéro de séquence 1, etc.

La sortie de trace des commandes est écrite avec un maximum de cinq fichiers de trace en boucle. La taille maximale de chaque fichier de trace est de 20MB.

Remarque : Si l'utilisateur exécutant la commande n'est pas autorisé à écrire dans le répertoire spécifié par le paramètre **-tracePath**, la sortie de trace est écrite dans l'erreur standard.

Exemple

Dans cet exemple, la commande **fteListAgents** est tracée et la trace est écrite dans le répertoire `/u/fteuser`:

```
./fteListAgents -trace com.ibm.wmqfte=all -tracePath /u/fteuser
```

Dans cet exemple, la commande **fteCreateTransfer** est tracée et la trace est écrite dans le répertoire `/tmp`:

```
./fteCreateTransfer -trace com.ibm.wmqfte=all -tracePath /tmp -t text -sa AGENT1 -da AGENT2 -df /tmp/IEEUJV.txt "'SYS1.SAMPLIB(IEEUJV)'"
```

Le fichier de trace écrit dans `/tmp` contient uniquement des informations sur le traitement effectué par la commande **fteCreateTransfer**, telles que la façon dont la commande génère le message de demande de transfert envoyé à l'agent et la durée pendant laquelle elle attend que l'agent renvoie un accusé de réception indiquant qu'il a reçu la demande. Le fichier de trace ne contient aucune information sur le transfert lui-même.

Pour collecter une trace d'une commande Managed File Transfer for z/OS soumise à l'aide de JCL, vous devez effectuer les étapes suivantes.

Procédure

1. Localisez le fichier contenant le JCL de la commande qui doit être tracée.
2. Dans le fichier, recherchez le membre de cette commande.
3. Editez le membre et localisez la ligne qui contient le nom de la commande à tracer. Modifiez cette ligne de sorte qu'elle inclue le texte après le nom de la commande et avant le signe + :

```
-trace classes=level -tracePath directory_path
```

où :

- *classes=level* est le niveau de trace à utiliser et les classes pour lesquelles activer la trace. Sauf indication contraire de votre représentant de support IBM , définissez ce paramètre sur `com.ibm.wmqfte=all`.
- *directory_path* est le répertoire USS dans lequel les fichiers de trace seront écrits.

4. Soumettez le membre.

5. Une fois le problème reproduit, éditez à nouveau le membre et supprimez le texte:

```
-trace classes=level -tracePath directory_path
```

que vous avez ajouté à l'étape «3», à la page 299.

Résultats

Les fichiers de trace résultants sont écrits dans le répertoire spécifié par le paramètre **-tracePath** .

Les fichiers de trace sont appelés `trace%PID%.txt.number`, où:

- *%PID%* est l'identificateur de processus de la commande.
- *numéro* est un numéro de séquence du fichier de trace. En règle générale, les informations de trace générées par une commande sont contenues dans un fichier de trace unique dont le numéro de séquence est 0.

Toutefois, il est possible qu'une commande génère beaucoup d'informations de trace. Dans ce cas, la trace sera écrite dans plusieurs fichiers. Le fichier de trace en cours a le numéro de séquence 0, le fichier de trace le plus ancien suivant a le numéro de séquence 1, etc.

La sortie de trace des commandes est écrite avec un maximum de cinq fichiers de trace en boucle. La taille maximale de chaque fichier de trace est de 20MB.

Remarque : Si l'utilisateur exécutant la commande n'est pas autorisé à écrire dans le répertoire spécifié par le paramètre **-tracePath** , la sortie de trace est écrite dans l'erreur standard.

Exemple

Dans cet exemple, le membre BFGMNLI a été modifié pour tracer la commande **ftelistmonitors** :

```

//*****
//* <copyright
//* notice="lm-source"
//* pids="5655-MF9"
//* years="2013,2016"
//* crc="3927276320" >
//* Licensed Materials - Property of IBM
//*
//* 5655-MF9
//*
//* (C) Copyright IBM Corp. 2013, 2023. All Rights Reserved.
//* </copyright>
//*****

```

```

/* fteListMonitors
/*****
//BFGCMD EXEC PGM=IKJEFT01,REGION=0M
//SYSEXEC DD DSN=++LIBRARY++,DISP=SHR
//SYSTSPT DD SYSOUT=*
//STDOUT DD SYSOUT=*
//STDERR DD SYSOUT=*
//SYSTSIN DD *
%BFGCMD CMD=fteListMonitors -trace com.ibm.wmqfte=all -tracePath /u/trace +
-v -p QM1
/*
//

```

Lorsque le membre est soumis, la commande **fteListMonitors** écrit la trace dans le répertoire USS /u/trace.

z/OS *Traçage des consigneurs de base de données autonome Managed File Transfer for z/OS*
 La fonction de trace de Managed File Transfer for z/OS est fournie pour aider le support IBM à diagnostiquer vos problèmes. Diverses commandes et propriétés contrôlent le comportement de cette fonction.

Pourquoi et quand exécuter cette tâche

Si vous êtes invité à fournir une sortie de trace pour examiner un problème avec un consigneur de base de données autonome, utilisez l'une des options suivantes.

Si vous ne savez pas quelle option utiliser, contactez votre représentant du support IBM qui vous conseillera sur la meilleure façon de collecter des données de trace pour le problème que vous rencontrez.

Procédure

- S'il est possible d'arrêter un consigneur pendant une courte période, collectez une trace du consigneur à partir du démarrage.
 Pour plus d'informations, voir [«Collecte d'une trace de consigneur de base de données autonome Managed File Transfer for z/OS à partir du démarrage»](#), à la page 300.
- S'il n'est pas possible d'arrêter un consigneur, collectez une trace de manière dynamique à l'aide de la commande **fteSetLoggerTraceLevel**.
 Pour plus d'informations, voir [«Collecte dynamique d'une trace de consigneur de base de données autonome Managed File Transfer for z/OS»](#), à la page 303.

z/OS *Collecte d'une trace de consigneur de base de données autonome Managed File Transfer for z/OS à partir du démarrage*

Lorsqu'il est possible d'arrêter un consigneur pendant une courte période, vous devez collecter la trace du consigneur IBM MQ Managed File Transfer depuis le démarrage.

Pourquoi et quand exécuter cette tâche

Le mode de collecte de la trace varie selon que le consigneur est administré à l'aide de UNIX System Services (USS) ou de JCL.

Si vous ne savez pas laquelle des options suivantes utiliser, contactez votre représentant du support IBM qui vous conseillera sur la meilleure façon de collecter des données de trace pour le problème que vous rencontrez.

Procédure

- Si vous utilisez:
 - UNIX System Services (USS), voir [«Collecte d'une trace de consigneur de base de données autonome à partir du démarrage à l'aide d'USS»](#), à la page 301.

- JCL, voir «Collecte d'une trace de consignateur de base de données autonome à partir du démarrage à l'aide de JCL», à la page 302.

z/OS Collecte d'une trace de consignateur de base de données autonome à partir du démarrage à l'aide d'USS

Pour collecter une trace d'un consignateur Managed File Transfer for z/OS qui est administré à l'aide des services système UNIX (USS) à partir du démarrage, vous devez définir diverses propriétés dans le fichier `logger.properties` pour ce consignateur avant de le démarrer.

Pourquoi et quand exécuter cette tâche

Pour collecter une trace à partir du démarrage, procédez comme suit.

Procédure

1. Localisez le fichier `logger.properties` du consignateur qui doit être tracé.

Le fichier `logger.properties` se trouve dans le répertoire `BFG_DATA/mqft/config/coordination_qmgr_name/loggers/logger_name`.

2. Editez le fichier et ajoutez des entrées pour les propriétés suivantes:

- **trace**=*spécification de trace*

La propriété **trace** détermine les classes internes et les packages à tracer. Sauf indication contraire de votre représentant du support IBM, définissez cette propriété sur la valeur `com.ibm.wmqfte=all`.

- **traceFiles**=*nombre de fichiers de trace à utiliser*
- **traceSize**=*taille de chaque fichier de trace, en Mo*

Les propriétés **traceFiles** et **traceSize** sont utilisées pour contrôler la quantité de données de trace collectées. Vous devez définir ces propriétés sur des valeurs importantes afin de collecter autant de données de trace que possible.

Par exemple, pour collecter 1GB de trace d'encapsulation à l'aide de la spécification de trace `com.ibm.wmqfte=all`, ajoutez les lignes suivantes au fichier `logger.properties` :

```
trace=com.ibm.wmqfte=all
traceFiles=5
traceSize=200
```

Ainsi, le consignateur écrit les données de trace dans un maximum de 5 fichiers, chaque fichier ayant une taille de 200MB.

Pour plus d'informations sur ces propriétés de consignateur, voir [Le fichier MFT logger.properties](#).

3. Arrêtez le consignateur à tracer à l'aide de la commande **fteStopLogger**.
4. Démarrez le consignateur en exécutant la commande **fteStartLogger**.
5. Reproduisez le problème.
6. Arrêtez le consignateur.
7. Editez le fichier `logger.properties` du consignateur et supprimez les entrées des propriétés **trace**, **traceFiles** et **traceSize** que vous avez ajoutées à l'étape «2», à la page 301.

Cela permet de s'assurer que la trace n'est pas activée lors du prochain redémarrage du consignateur.

Résultats

Les fichiers de trace résultants sont écrits dans le répertoire `BFG_DATA/mqft/logs/coordination_qmgr_name/loggers/logger_name/logs/trace%PID%`, où `%PID%` est l'identificateur de processus du consignateur.

Collecte d'une trace de consignateur de base de données autonome à partir du démarrage à l'aide de JCL

Pour collecter une trace d'un consignateur Managed File Transfer for z/OS qui est administré à l'aide de JCL à partir du démarrage, vous devez définir diverses propriétés dans le fichier `logger.properties` pour ce consignateur avant de le démarrer.

Pourquoi et quand exécuter cette tâche

Pour collecter une trace à partir du démarrage, procédez comme suit.

Procédure

1. Localisez le fichier `logger.properties` du consignateur qui doit être tracé.

Le fichier `logger.properties` se trouve dans le répertoire `BFG_DATA/mqft/config/coordination_qmgr_name/loggers/logger_name`.

2. Editez le fichier et ajoutez des entrées pour les propriétés suivantes:

- **trace**=spécification de trace

La propriété **trace** détermine les classes internes et les packages à tracer. Sauf indication contraire de votre représentant du support IBM, définissez cette propriété sur la valeur `com.ibm.wmqfte=all`.

- **traceFiles**=nombre de fichiers de trace à utiliser
- **traceSize**=taille de chaque fichier de trace, en Mo

Les propriétés **traceFiles** et **traceSize** sont utilisées pour contrôler la quantité de données de trace collectées. Vous devez définir ces propriétés sur des valeurs importantes afin de collecter autant de données de trace que possible.

Par exemple, pour collecter 1GB de trace d'encapsulation à l'aide de la spécification de trace `com.ibm.wmqfte=all`, ajoutez les lignes suivantes au fichier `logger.properties` :

```
trace=com.ibm.wmqfte=all
traceFiles=5
traceSize=200
```

Ainsi, le consignateur écrit les données de trace dans un maximum de 5 fichiers, chaque fichier ayant une taille de 200MB.

Pour plus d'informations sur ces propriétés de consignateur, voir [Le fichier MFT logger.properties](#).

3. Localisez le fichier contenant le JCL du consignateur qui doit être tracé.
 4. Soumettez le membre `BFGLSPG` dans le fichier pour arrêter le consignateur.
 5. Redémarrez le consignateur en soumettant le membre `BFGLGST` dans le fichier.
 6. Reproduisez le problème.
 7. Soumettez le membre `BFGLSPG` dans le fichier pour arrêter à nouveau le consignateur.
 8. Editez le fichier `logger.properties` du consignateur et supprimez les entrées des propriétés **trace**, **traceFiles** et **traceSize** que vous avez ajoutées à l'étape «2», à la page 302.
- Cela permet de s'assurer que la trace n'est pas activée lors du prochain redémarrage du consignateur.

Résultats

Les fichiers de trace résultants sont écrits dans le répertoire `BFG_DATA/mqft/logs/coordination_qmgr_name/loggers/logger_name/logs/trace%PID%`, où `%PID%` est l'identificateur de processus du consignateur.

Collecte dynamique d'une trace de consignateur de base de données autonome Managed File Transfer for z/OS

Lorsqu'il n'est pas possible d'arrêter un consignateur pendant une courte période, vous devez collecter la trace du consignateur Managed File Transfer for z/OS de manière dynamique.

Pourquoi et quand exécuter cette tâche

Le mode de collecte de la trace varie selon que le consignateur est administré à l'aide des services système UNIX (USS) ou du JCL.

Si vous ne savez pas laquelle des options suivantes utiliser, contactez votre représentant du support IBM qui vous conseillera sur la meilleure façon de collecter des données de trace pour le problème que vous rencontrez.

Procédure

- Si vous utilisez:
 - Unix System Services (USS), voir «Collecte dynamique d'une trace de consignateur de base de données autonome à l'aide d'USS», à la page 303.
 - JCL, voir «Collecte dynamique d'une trace de consignateur de base de données autonome à l'aide de JCL», à la page 304.

 Collecte dynamique d'une trace de consignateur de base de données autonome à l'aide d'USS
Sous (USS), vous pouvez utiliser la commande **fteSetLoggerTraceLevel** pour collecter la trace d'un consignateur en cours d'exécution. Cela peut être très utile si le support IBM a besoin de voir une trace d'un consignateur qui ne peut pas être arrêtée.

Pourquoi et quand exécuter cette tâche

Pour collecter une trace à partir d'un consignateur Managed File Transfer for z/OS à l'aide de la commande **fteSetLoggerTraceLevel**, procédez comme suit.

Procédure

1. Activez la trace pour le consignateur en exécutant la commande suivante:

```
fteSetLoggerTraceLevel -traceLogger classes=level logger_name
```

Le paramètre `-traceLogger` détermine les classes internes et les packages à tracer. Sauf indication contraire de votre représentant du support IBM, définissez cette propriété sur la valeur `com.ibm.wmqfte=all`.

2. Reproduisez le problème.
3. Désactivez la trace pour le consignateur en exécutant la commande suivante:

```
fteSetLoggerTraceLevel -traceLogger =off logger_name
```

4. Si un consignateur est occupé, les fichiers de trace peuvent s'encapsuler rapidement et écraser les informations nécessaires pour identifier le problème.

Si vous pouvez arrêter le consignateur pendant une courte période, procédez comme suit pour réduire la quantité de données de trace collectées. Sinon, contactez le support IBM et discutez des autres spécifications de trace pour réduire la quantité de données de trace collectées.

- a) Planifiez un certain temps pour arrêter le consignateur.
- b) Localisez le fichier `logger.properties` du consignateur qui doit être tracé.

Le fichier `logger.properties` se trouve dans le répertoire `BFG_DATA/mqft/config/coordination_qmgr_name/loggers/logger_name`.

c) Editez le fichier et ajoutez des entrées pour les propriétés suivantes:

```
traceFiles=nombre_fichiers_trace_à_utiliser  
traceSize=taille_du_fichier_trace_en_Mo
```

Les propriétés **traceFiles** et **traceSize** sont utilisées pour contrôler la quantité de données de trace collectées.

La valeur par défaut de la propriété **traceFiles** est 5 et celle de la propriété **traceSize** est 20MB. Cela signifie que si vous activez la trace de manière dynamique et que vous n'avez pas défini les propriétés, l'agent écrit les informations de trace dans 5 fichiers de trace d'encapsulation, chacun d'eux ayant une taille maximale de 20MB.

Vous devez définir ces propriétés sur des valeurs importantes afin de collecter autant de données de trace que possible.

Par exemple, pour collecter 1GB de trace d'encapsulation, ajoutez les lignes suivantes au fichier `logger.properties` :

```
traceFiles=5  
traceSize=200
```

Ainsi, le consignateur écrit les données de trace dans un maximum de 5 fichiers, chaque fichier ayant une taille de 200MB.

Pour plus d'informations sur ces propriétés de consignateur, voir [Le fichier MFT logger.properties](#).

- d) Arrêtez le consignateur en exécutant la commande **fteStopLogger** .
- e) Démarrez le consignateur en exécutant la commande **fteStartLogger** .
- f) Activez la trace pour le consignateur en exécutant la commande suivante:

```
fteSetLoggerTraceLevel -traceLogger trace_specification logger_name
```

- g) Reproduisez le problème.
- h) Désactivez la trace pour le consignateur en exécutant la commande suivante:

```
fteSetLoggerTraceLevel -traceLogger =off logger_name
```

Résultats

Les fichiers de trace sont écrits dans le répertoire `BFG_DATA/mqft/logs/coordination_qmgr_name/loggers/logger_name/logs/trace%PID%`, où `%PID%` est l'identificateur de processus du consignateur.

 *Collecte dynamique d'une trace de consignateur de base de données autonome à l'aide de JCL*

Vous pouvez utiliser le membre `BFGLGTC` dans le fichier contenant le JCL, pour le consignateur qui doit être tracé, afin de collecter la trace à partir d'un consignateur Managed File Transfer for z/OS en cours d'exécution. Cela peut être très utile si le support IBM a besoin de voir une trace d'un consignateur qui ne peut pas être arrêté.

Pourquoi et quand exécuter cette tâche

Pour collecter une trace à partir d'un consignateur à l'aide du membre `BFGLGTC`, procédez comme suit.

Procédure

1. Localisez le fichier contenant le JCL du consignateur qui doit être tracé.
2. Editez le membre `BFGLGTC` dans le jeu de données et localisez la ligne qui contient le texte:

```
-traceLogger
```

Le texte qui suit contient la liste des classes internes et des packages à tracer. Par défaut, cette liste est définie sur:

```
com.ibm.wmqfte=all
```

Sauf indication contraire de votre représentant du support IBM, conservez cette valeur telle qu'elle est.

3. Soumettez le membre BFGLGTC.
4. Reproduisez le problème.
5. Editez à nouveau le membre BFGLGTC et définissez le paramètre **-traceLogger** sur *= off*, comme suit:

```
-traceLogger =off +
```

6. Soumettez à nouveau le membre BFGLGTC pour désactiver la trace.
7. Si un consignateur est occupé, il est possible que les fichiers de trace soient encapsulés rapidement et écrasent les informations nécessaires pour identifier le problème.

Dans cette situation, il existe deux options:

a) La première option consiste à:

- i) Planifiez un certain temps pour arrêter le consignateur.
- ii) Localisez le fichier `logger.properties` du consignateur qui doit être tracé. Le fichier `logger.properties` se trouve dans le répertoire `BFG_DATA/mqft/config/coordination_qmgr_name/loggers/logger_name`.
- iii) Editez le fichier et ajoutez des entrées pour les propriétés suivantes:

- **traceFiles**=nombre de fichiers de trace à utiliser
- **traceSize**=taille de chaque fichier de trace, en Mo

Les propriétés **traceFiles** et **traceSize** sont utilisées pour contrôler la quantité de données de trace collectées.

La valeur par défaut de la propriété **traceFiles** est 5 et celle de la propriété **traceSize** est 20MB. Cela signifie que si vous activez la trace de manière dynamique et que vous n'avez pas défini les propriétés, le consignateur écrit les informations de trace dans 5 fichiers de trace d'encapsulation, chacun ayant une taille maximale de 20MB.

Vous devez définir ces propriétés sur des valeurs importantes afin de collecter autant de données de trace que possible.

Par exemple, pour collecter 1GB de trace d'encapsulation, ajoutez les lignes suivantes au fichier `logger.properties` :

```
traceFiles=5  
traceSize=200
```

Ainsi, le consignateur écrit les données de trace dans un maximum de 5 fichiers, chaque fichier ayant une taille de 200MB.

Pour plus d'informations sur ces propriétés de consignateur, voir [Le fichier MFT `logger.properties`](#).

- iv) Localisez le fichier contenant le JCL du consignateur qui doit être tracé.
- v) Soumettez le membre BFGLSPG dans le fichier pour arrêter le consignateur.
- vi) Redémarrez le consignateur en soumettant le membre BFGLGST dans le fichier.
- vii) Editez le membre BFGLGTC dans le fichier et localisez la ligne qui contient le texte:

```
-traceLogger
```

Le texte qui suit contient la liste des classes internes et des packages à tracer. Par défaut, cette liste est définie sur:

```
com.ibm.wmqfte=all
```

Sauf indication contraire de votre représentant du support IBM, conservez cette valeur telle qu'elle est.

- viii) Lorsqu'il est temps d'activer la trace, soumettez le membre BFGLGTC.
- ix) Reproduisez le problème.
- x) Editez à nouveau le membre BFGLGTC et définissez le paramètre **-traceLogger** sur `= off`, comme suit:

```
-traceLogger =off +
```

- xi) Soumettez à nouveau le membre BFGLGTC pour désactiver la trace.
- b) La deuxième option consiste à contacter votre représentant du support IBM s'il n'est pas possible d'arrêter le consignateur pendant une courte période.
- Vous pouvez ensuite discuter d'autres spécifications de trace à utiliser afin de réduire la quantité de données de trace générées.

Résultats

Les fichiers de trace résultants sont écrits dans le répertoire `BFG_DATA/mqft/logs/coordination_qmgr_name/loggers/logger_name/logs/trace%PID%`, où `%PID%` est l'identificateur de processus du consignateur.

Problèmes MFT courants

Problèmes courants susceptibles de se produire dans votre réseau Managed File Transfer .

- Si un transfert de texte échoue avec l'erreur suivante:

```
BFGI00060E: Echec de la conversion des données texte
```

Cela peut se produire pour l'une des deux raisons suivantes:

1. Un ou plusieurs caractères du fichier source ne peuvent pas être convertis de la page de codes du fichier source vers la page de codes du fichier cible. Ce problème peut se produire lorsque les pages de codes ont des jeux de caractères différents et que certains caractères ne peuvent pas être convertis entre eux.

S'il est acceptable que la conversion de certains caractères ne soit pas convertie, une séquence de caractères de remplacement peut être définie au niveau de l'agent de destination afin que le transfert n'échoue pas. Spécifiez la propriété d'agent **textReplacementCharacterSequence** pour définir une séquence de caractères de remplacement. Pour plus d'informations, voir [Tableau 45](#), à la page 566.

2. Le codage du fichier source ne correspond pas au codage par défaut de l'agent source. Dans ce cas, l'exécution d'un transfert de texte à l'aide des paramètres par défaut endommage les données de type caractères. Pour transférer un fichier source dont le codage n'est pas identique à celui de l'agent source, effectuez l'une des opérations suivantes:
 - a. Indiquez le codage de fichier dans un fichier de définition de transfert. Pour plus d'informations, voir la section «[Utilisation de fichiers de définition de transfert](#)», à la page 164.
 - b. Spécifiez le codage de fichier à l'aide du paramètre **-sce** avec la commande **fteCreateTransfer** . Pour plus d'informations, voir «[fteCreateTransfer: démarrer un nouveau transfert de fichier](#)», à la page 447.
 - c. Spécifiez le codage de fichier dans le cadre d'une tâche de déplacement ou de copie Ant. Pour plus d'informations, voir la section «[Utilisation d' Apache Ant avec MFT](#)», à la page 258.

Pour vérifier que vous avez sélectionné le codage de fichier source correct pour un transfert, procédez comme suit:

1. Définissez le codage du fichier de destination sur UTF-8.
 2. Transférez le fichier en mode texte.
 3. Utilisez un afficheur de fichiers UTF-8 pour afficher le contenu du fichier. Si tous les caractères du fichier sont affichés correctement, le codage du fichier source est correct.
- Si vous voyez la sortie suivante de la commande **fteCreateAgent** :

```
BFGMQ1007I: Le gestionnaire de files d'attente de coordination ne peut pas être contacté ou a
refusé un
connection attempt.
Le code anomalie IBM MQ était 2058. La présence de l'agent ne sera pas publiée.
```

Indique que le gestionnaire de files d'attente de coordination ne peut pas être contacté et fournit le code anomalie IBM MQ pour la raison. Ce message d'information peut indiquer que le gestionnaire de files d'attente de coordination est actuellement indisponible ou que vous avez défini la configuration de manière incorrecte.

- Si vous utilisez des routines d'exit utilisateur et qu'un incident se produit lors de l'appel de l'exit utilisateur ou juste après l'appel de l'exit, par exemple une panne de produit ou une coupure de courant, il est possible que l'exit utilisateur soit appelé plusieurs fois.
- Si vous disposez d'un agent avec un gestionnaire de files d'attente sur un système avec une adresse IP affectée par DHCP (et non une adresse IP statique), *et* que l'agent se connecte à ce système à l'aide d'une connexion TCP/IP client, vous devez démarrer l'agent avec la variable d'environnement système suivante définie:

–  Sur Windows :

```
set BFG_JVM_PROPERTIES="-Dsun.net.inetaddr.ttl=value"
```

–  Sur UNIX :

```
export BFG_JVM_PROPERTIES="-Dsun.net.inetaddr.ttl=value"
```

où *value* est l'intervalle de temps, en secondes, entre chaque vidage des valeurs DNS en cache de la machine virtuelle Java. Si l'adresse IP du système de gestionnaire de files d'attente est réaffectée pour une raison quelconque (par exemple, en raison d'une indisponibilité du réseau, d'une expiration du bail IP ou d'un réamorçage du système), l'agent signale la perte de sa connexion au gestionnaire de files d'attente. Une fois le cache DNS de la machine virtuelle Java vidé, l'agent peut se reconnecter. Si cette variable d'environnement n'est pas définie, l'agent ne peut pas se reconnecter dans ce scénario sans redémarrer la machine virtuelle Java. Ce comportement est dû au fait que la machine virtuelle Java met en cache en interne les adresses IP des noms d'hôte et ne les actualise pas par défaut.

- Si vous exécutez la commande **fteStartAgent** et que le message d'erreur suivant s'affiche, votre environnement contient probablement des chemins d'accès à la bibliothèque supplémentaires qui entrent en conflit avec Managed File Transfer :

```
BFGCL0001E: Une erreur interne s'est produite. Exception : 'CC=2;RC=2495;AMQ8568:
La bibliothèque JNI native « mqjbnd » est introuvable. [3=mqjbnd]
```

Si la variable d'environnement LD_LIBRARY_PATH ou LIBPATH est définie pour référencer une version 64 bits de la bibliothèque avant la version 32 bits lorsque l'agent est exécuté avec une version 32 bits de Java (comme c'est actuellement le cas de la plupart des plateformes), cette erreur se produit.

Pour résoudre ce problème, définissez la propriété javaLibraryPath de l'agent Managed File Transfer afin qu'elle référence l'emplacement correct de la bibliothèque. Par exemple, pour mqjbnd sur AIX, définissez la valeur sur : /usr/mqm/java/lib. Pour mqjbnd sous Linux, définissez sur : /opt/mqm/java/lib

- Si vous avez activé la vérification des droits d'accès utilisateur en spécifiant `authorityChecking=true` dans le fichier de propriétés de l'agent et que toutes les vérifications des droits d'accès échouent même si l'utilisateur dispose des droits d'accès requis dans la file d'attente des droits d'accès appropriés:
 - Vérifiez que l'utilisateur qui exécute l'agent dispose du contrôle d'accès ALT_USER sur le gestionnaire de files d'attente de l'agent.

- Si vous avez activé la vérification des droits utilisateur en spécifiant `authorityChecking=true` dans le fichier de propriétés de l'agent et que des messages d'erreur IBM MQ sont écrits dans le fichier `output0.log` de l'agent, effectuez l'une des actions suivantes:
 - Ignorez les messages, l'agent n'est pas affecté.
 - Accordez à l'utilisateur qui exécute le droit GET d'agent sur `SYSTEM.FTE.AUTH*` appartenant à l'agent.
- Si vous avez modifié le fichier de propriétés de l'agent et que l'agent ne les a pas pris en compte:
 - Redémarrez l'agent pour vous assurer qu'il lit les nouvelles propriétés.

z/OS



- Si vous utilisez l'agent sous z/OS pour effectuer un transfert vers un fichier PDS ou PDSE et qu'une fin anormale se produit, l'espace disque de votre système peut être limité. La fin anormale est probablement associée au code achèvement système B14 avec le code retour 0C, indiquant qu'il ne reste plus d'espace.

Si vous effectuez un transfert vers un fichier séquentiel, le transfert échoue et indique la condition de manque d'espace, mais l'agent reste opérationnel.

- Si vous utilisez l'agent sous z/OS et que la tâche WMQFTEP génère des vidages core Java avant de ne plus répondre, appliquez l'APAR OA43472 des services système OMVS.
- Si la sortie suivante s'affiche lors de l'exécution d'un script de configuration ou d'administration sous z/OS:

```
Erreur de syntaxe FSUM7332 : obtenu (, Newline attendu
```

Cette sortie indique que la variable d'environnement `_BPXK_AUTOCVT=ON` n'a pas été définie dans l'environnement dans lequel le script de configuration ou d'administration est exécuté. Pour plus d'informations sur cette variable d'environnement et pour savoir comment la définir, voir «[Variables d'environnement pour MFT sous z/OS](#)», à la page 88.

Problèmes MFT courants avec JZOS

Voici quelques suggestions si vous rencontrez des problèmes avec JZOS.

- Si le traitement de JZOS échoue:
 - Ajoutez `PARM= '+T'` au JCL. Exemple :

```
//MQMFT EXEC PGM=JVMLDM86,REGION=0M,PARM= '+T'
```

- Ajoutez `set -x` au fichier d'environnement

- Si vous obtenez:

```
JVMJZBL1038E Le processus shell enfant s'est arrêté avec le code de sortie: 1
JVMJZBL1042E Echec du lanceur par lots JZOS, retour code=102
```

Cela signifie qu'il y a eu un problème avec votre fichier d'environnement et les commandes Managed File Transfer . Cela peut être dû à des chemins non valides spécifiés.

- Dans votre fichier d'environnement, recherchez la valeur de **BFG_PROD**.

1. Accédez à OMVS et utilisez la commande **ls -ltr**.

Par exemple, si **BFG_PROD** est `/HMF8800/`, entrez la commande suivante:

```
ls -ltr HMF8800/bin/fteBatch
```

2. Vérifiez que ce fichier existe et que le travail par lots dispose des droits de lecture sur le fichier.
3. Résolvez les problèmes éventuels.

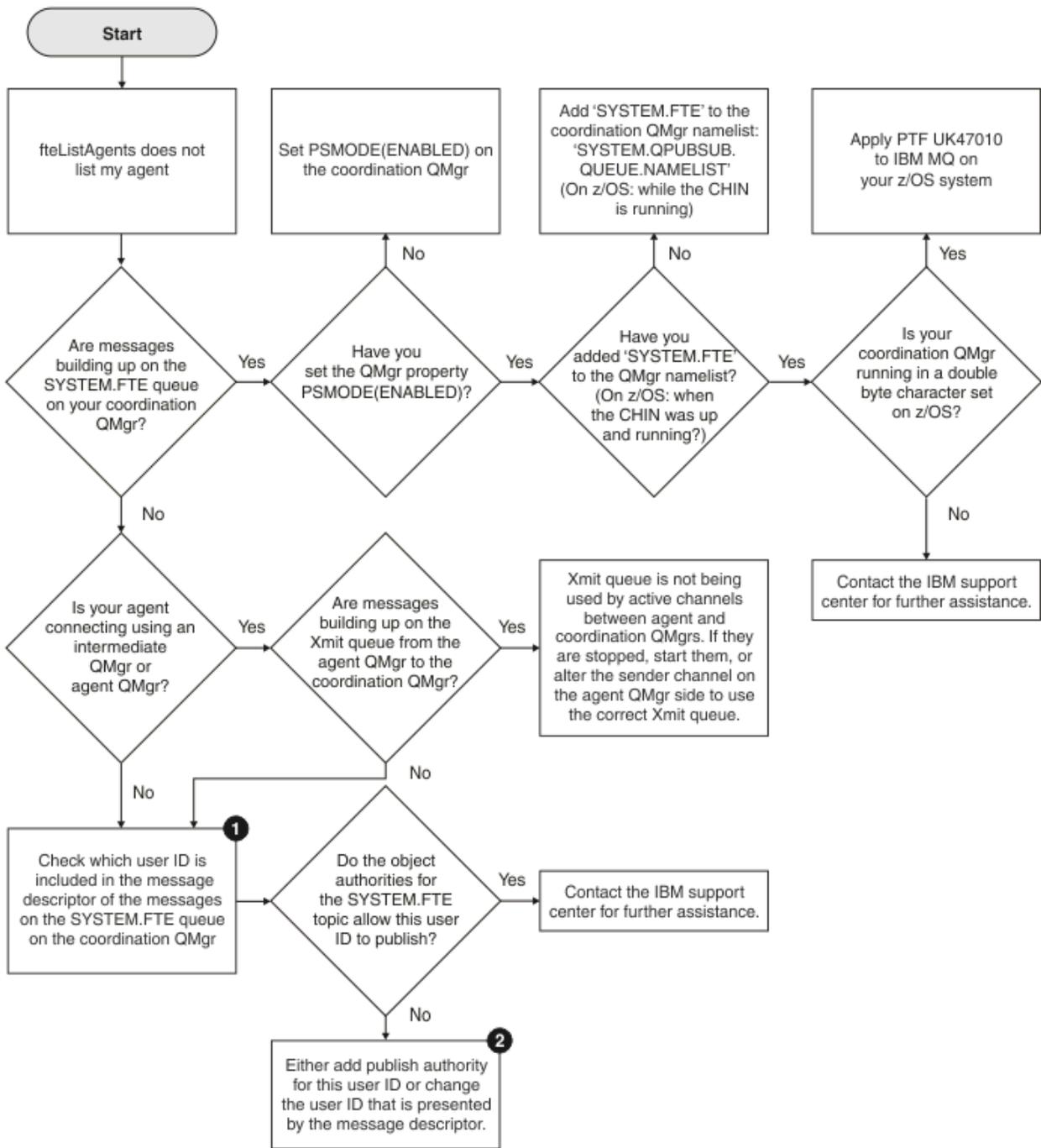
- Si le JCL ne parvient toujours pas à s'exécuter correctement:

1. Créez un fichier dans USS, par exemple, myenv et utilisez un éditeur pour copier les informations du fichier d'environnement dans ce fichier myenv .
2. Sauvegardez ce fichier.
3. A partir de la ligne de commande, utilisez la commande **chmod +x myenv** pour autoriser l'exécution du fichier.
4. Exécutez la commande **. myenv**. Notez qu'il s'agit de (nom de fichier vide).
L'exécution de cette commande signale toute erreur dans le fichier myenv .
5. Corrigez les erreurs dans les fichiers myenv et d'environnement.

Que faire si votre agent MFT n'est pas répertorié par la commande `fteListAgents` ?

Si votre agent n'est pas répertorié par la commande **fteListAgents** ou n'est pas affiché dans IBM MQ Explorer, ou si vos transferts de fichier ne sont pas affichés dans le **journal de transfert** de IBM MQ Explorer, vous pouvez effectuer un certain nombre d'étapes afin d'identifier le problème.

Servez-vous du diagramme suivant pour diagnostiquer les problèmes et décider de l'action à effectuer :



Clé du diagramme :

1. Pour savoir comment vérifier l'ID utilisateur qui est présenté, voir «[Examen des messages avant la publication](#)», à la [page 326](#). Les ID utilisateur doivent respecter le nombre limite de caractères des noms d'utilisateur MQ, qui est de 12. Si un nom d'utilisateur comporte plus de 12 caractères (Administrator, par exemple), le nom d'utilisateur sera tronqué avant d'être vérifié pour l'autorisation. Dans un exemple utilisant Administrator, le message d'erreur suivant est ajouté au journal des erreurs du gestionnaire de files d'attente:

AMQ8075: Authorization failed because the SID for entity 'administrato' cannot be obtained.

2. Pour plus d'informations sur les droits requis pour la file d'attente SYSTEM.FTE, voir «Droits de publication du journal et des messages d'état des agents MFT», à la page 373.

UNIX Que faire si votre processus d'agent MFT disparaît mais qu'aucune information de diagnostic n'est consignée

Sur les plateformes UNIX , si un processus d'agent a disparu mais que les fichiers journaux de l'agent ne contiennent aucune explication, cela peut être dû à la manière dont l'agent a été démarré.

Vous pouvez vérifier les informations de diagnostic de l'agent de l'une des manières suivantes:

- Vérifiez si les fichiers journaux de l'agent indiquent que l'agent a été arrêté.
- Vérifiez si le fichier de verrouillage d'agent `agent.lock` existe toujours.

Si vous démarrez l'agent à partir d'un script shell par exemple, tous les processus enfant associés à ce script sont supprimés à la fin du script (y compris le processus d'agent). Pour que l'agent continue de s'exécuter au-delà de la durée du script qui l'a appelé, procédez comme suit:

1. Préfixez la commande **fteStartAgent** avec la commande **nohup** pour dissocier le processus **fteStartAgent** (et tous les processus enfant) du script.

A l'avenir, lorsque le script s'arrêtera, l'agent continuera de s'exécuter.

Que faire si un agent est affiché comme étant à l'état UNKNOWN (inconnu)

Votre agent est en cours d'exécution et répond correctement à la commande **ftePingAgent** , et les éléments sont transférés normalement. Toutefois, les commandes **fteListAgents** et **fteShowAgentDetails** , ainsi que le plug-in IBM MQ Explorer Managed File Transfer , indiquent que l'agent est à l'état UNKNOWN .

Pourquoi ce problème se produit

Chaque agent publie régulièrement son statut dans SYSTEM.FTE sur le gestionnaire de file d'attente de coordination. La fréquence à laquelle un agent publie son statut est contrôlée par les propriétés d'agent suivantes:

agentStatusPublishRateLimit

Fréquence maximale, en secondes, à laquelle l'agent republie son statut en raison d'un changement de statut de transfert de fichier. La valeur par défaut de cette propriété est de 30 secondes.

agentStatusPublishRateMin

Fréquence minimale, en secondes, à laquelle l'agent publie son statut. Cette valeur doit être supérieure ou égale à la valeur de la propriété **agentStatusPublishRateLimit** . La valeur par défaut de la propriété **agentStatusPublishRateMin** est 300 secondes (ou 5 minutes).

Les commandes **fteListAgents** et **fteShowAgentDetails** , ainsi que le plug-in IBM MQ Explorer Managed File Transfer (MFT), utilisent ces publications pour déterminer le statut d'un agent. Pour ce faire, les commandes et le plug-in effectuent les étapes suivantes:

1. Connectez-vous au gestionnaire de files d'attente de coordination.
2. Abonnez-vous à SYSTEM.FTE .
3. Recevoir les publications de statut de l'agent.
4. Créez une file d'attente temporaire sur le gestionnaire de file d'attente de coordination.
5. Placez un message dans la file d'attente temporaire et sauvegardez l'heure d'insertion afin d'obtenir l'heure en cours sur le système du gestionnaire de files d'attente de coordination.
6. Fermez la file d'attente temporaire.
7. Utilisez les informations contenues dans les publications, ainsi que l'heure en cours, pour déterminer le statut d'un agent.
8. Déconnectez-vous du gestionnaire de files d'attente de coordination.

Le message de statut d'un agent est considéré comme périmé si la différence entre l'heure de sa publication et l'heure en cours est supérieure à: *La valeur de la propriété d'agent **agentStatusPublishRateMin** (incluse dans le message de statut) plus la valeur de la propriété du gestionnaire de files d'attente de coordination avancée **agentStatusJitterTolerance**.*

Par défaut, la propriété **agentStatusJitterTolerance** a une valeur de 3000 millisecondes (3 secondes).

Si les propriétés **agentStatusPublishRateMin** et **agentStatusJitterTolerance** sont définies sur leurs valeurs par défaut, le statut d'un agent est considéré comme périmé si la différence entre l'heure à laquelle il a été publié et l'heure en cours est supérieure à 303 secondes (ou 5 minutes 3 secondes).

Tout agent avec un message de statut périmé est signalé par les commandes **fteListAgents** et **fteShowAgentDetails**, ainsi que par le plug-in IBM MQ Explorer MFT, comme étant à l'état UNKNOWN.

La publication de statut d'un agent peut être périmée pour l'une des raisons suivantes:

1. Il existe une différence significative dans l'heure système entre le système sur lequel s'exécute le gestionnaire de files d'attente d'agent et le système sur lequel se trouve le gestionnaire de files d'attente de coordination.
2. Les canaux entre le gestionnaire de files d'attente d'agent et le gestionnaire de files d'attente de coordination sont arrêtés (ce qui empêche les nouveaux messages d'état d'atteindre le gestionnaire de files d'attente de coordination).
3. Un problème d'autorisation empêche l'agent de publier son statut dans SYSTEM.FTE sur le gestionnaire de file d'attente de coordination.
4. Une défaillance de l'agent s'est produite.

Identification et résolution des problèmes

Il existe un certain nombre d'étapes à effectuer pour déterminer la raison pour laquelle le statut d'un agent est signalé comme étant UNKNOWN:

1. Vérifiez si l'agent est en cours d'exécution en vous connectant au système de l'agent. Si l'agent est arrêté, recherchez pourquoi il n'est plus en cours d'exécution. Une fois qu'il est à nouveau en cours d'exécution, vérifiez si son statut est signalé correctement.
2. Vérifiez que le gestionnaire de file d'attente de coordination est en cours d'exécution. Si ce n'est pas le cas, redémarrez-le, puis utilisez la commande **fteListAgents** ou **fteShowAgentDetails**, ou le plug-in IBM MQ Explorer MFT, pour voir si le statut de l'agent est désormais correctement signalé.
3. Si l'agent et les gestionnaires de files d'attente de coordination sont en cours d'exécution, consultez leurs journaux d'erreurs pour voir s'il existe des problèmes d'autorisation qui empêchent l'agent de publier ses messages d'état. Si les journaux indiquent que des problèmes d'autorisation se produisent, vérifiez que l'utilisateur exécutant le processus d'agent dispose des droits appropriés pour publier des messages dans SYSTEM.FTE sur le gestionnaire de file d'attente de coordination.

Si les journaux d'erreurs du gestionnaire de files d'attente ne signalent aucun problème d'autorisation, vérifiez que les messages d'état ne sont pas bloqués sur le réseau IBM MQ. Vérifiez que tous les canaux émetteur et récepteur utilisés pour acheminer les messages du gestionnaire de files d'attente d'agent vers le gestionnaire de files d'attente de coordination sont en cours d'exécution.

Si les canaux sont en cours d'exécution, vérifiez les files d'attente de transmission associées aux canaux pour vous assurer que les messages d'état ne sont pas bloqués dessus. En outre, vous devez vérifier les files d'attente de rebut pour les gestionnaires de files d'attente afin de vous assurer que les messages d'état n'y ont pas été placés pour une raison quelconque.

4. Si les canaux sont en cours d'exécution et que les messages d'état transitent par le réseau IBM MQ, l'élément suivant à vérifier est que le moteur de publication / abonnement en file d'attente du gestionnaire de files d'attente récupère les messages.

La commande **fteSetupCoordination**, qui est utilisée pour définir le gestionnaire de files d'attente de coordination, fournit des commandes MQSC qui doivent être exécutées sur le gestionnaire de files

d'attente de coordination pour configurer le moteur de publication / abonnement en file d'attente afin de recevoir des publications. Ces commandes effectuent les étapes suivantes:

- Créez le système SYSTEM.FTE et sa chaîne de rubrique associée.
- Définissez une file d'attente locale appelée SYSTEM.FTE qui sera utilisé pour recevoir des messages d'état entrants.
- Activez le moteur de publication / abonnement en file d'attente en définissant l'attribut **PSMODE** sur le gestionnaire de files d'attente sur ENABLED.
- Modifiez le système SYSTEM.QPUBSUB.QUEUE.NAMELIST , utilisée par le moteur de publication / abonnement en file d'attente, inclut une entrée pour le nouveau système SYSTEM.FTE , file d'attente.

Pour plus d'informations à ce sujet, y compris les commandes MQSC qui doivent être exécutées, voir «[fteSetupCoordination](#)», à la page 527.

S'il existe des messages dans SYSTEM.FTE , vous devez vérifier que SYSTEM.QPUBSUB.QUEUE.NAMELIST a été configurée correctement et contient une entrée pour cette file d'attente. Si l'entrée est manquante, le moteur de publication / abonnement en file d'attente ne détecte aucun message d'état entrant de l'agent et ne les traite pas.

Vous devez également vous assurer que l'attribut **PSMODE** du gestionnaire de files d'attente est défini sur ENABLED, qui active le moteur de publication / abonnement en file d'attente.

5. Si les canaux sont en cours d'exécution et que les messages d'état transitent par le IBM MQ réseau et sont extraits de SYSTEM.FTE file d'attente par le moteur de publication / abonnement en file d'attente du gestionnaire de files d'attente, puis collectez les traces suivantes:

- Une trace IBM MQ MFT de l'agent, couvrant une période égale à trois fois la valeur de la propriété d'agent **agentStatusPublishRateMin**. Cela garantit que la trace couvre le moment où l'agent publie au moins trois messages contenant son statut. La trace doit être collectée de manière dynamique, à l'aide de la spécification de trace:

```
com.ibm.wmqfte.statestore.impl.FTEAgentStatusPublisher,  
com.ibm.wmqfte.utils.AgentStatusDetails,  
com.ibm.wmqfte.wmqiface.AgentPublicationUtils,  
com.ibm.wmqfte.wmqiface.RFHMessageFactory=all
```

Remarque : Une quantité réduite de trace est générée à l'aide de ces chaînes.

Pour plus d'informations sur l'activation de la trace pour les agents s'exécutant sous IBM MQ for Multiplatforms, voir «[Collecte dynamique d'une trace d'agent Managed File Transfer](#)», à la page 286.

Pour plus d'informations sur l'activation de la trace pour les agents s'exécutant sous IBM MQ for z/OS, voir «[Collecte dynamique d'une trace d'agent Managed File Transfer for z/OS](#)», à la page 294.

- Trace simultanée des gestionnaires de files d'attente utilisée pour acheminer les messages d'état du gestionnaire de files d'attente d'agent vers le gestionnaire de files d'attente de coordination.
- Trace de la commande **fteListAgents** , couvrant l'heure à laquelle l'agent est affiché comme étant à l'état UNKNOWN. La trace doit être collectée à l'aide de la spécification de trace:

```
com.ibm.wmqfte=all
```

Pour plus d'informations sur l'activation de la trace pour les commandes s'exécutant sous IBM MQ for Multiplatforms, voir «[Traçage des commandes Managed File Transfer sur Multiplatforms](#)», à la page 287.

Pour plus d'informations sur l'activation de la trace pour les commandes s'exécutant sous IBM MQ for z/OS, voir «[Commandes Managed File Transfer for z/OS de traçage](#)», à la page 297.

Une fois les traces collectées, elles doivent être mises à la disposition du support IBM pour analyse.

Information associée

[fteListAgents](#)

[fteShowAgentDetails](#)

Valeurs de statut de l'agent MFT

Le fichier MFT agent.properties

Le fichier MFT coordination.properties

Que faire si `ftePingAgent` dépasse le délai d'attente et signale un message BFGCL0214I

ftePingAgent est un utilitaire de ligne de commande utile fourni avec IBM MQ Managed File Transfer qui vous permet de vérifier si un agent est accessible et s'il peut répondre aux demandes.

Fonctionnement de la commande

Vous pouvez utiliser la commande **ftePingAgent** pour vérifier si un agent est accessible et s'il peut traiter des demandes. Lorsque la commande est exécutée, elle effectue les étapes suivantes:

- Se connecte au gestionnaire de files d'attente de commandes pour la topologie Managed File Transfer (MFT).
- Crée une file d'attente de réponses temporaire sur le gestionnaire de files d'attente de commandes.

Par défaut, la file d'attente temporaire a un nom qui commence par le préfixe WMQFTE. Toutefois, vous pouvez le modifier en définissant la propriété **dynamicQueuePrefix** dans le fichier MFT command.properties de l'installation.

- Envoie un message de demande d'agent MFT à la file d'attente SYSTEM.FTE.COMMAND.*nom_agent* sur le gestionnaire de files d'attente d'agent, via le gestionnaire de files d'attente de commandes. Le message de demande contient le nom de la file d'attente de réponses temporaire.
- Attend qu'un message de réponse arrive dans la file d'attente de réponses temporaire.

L'une des unités d'exécution au sein d'un agent est `CommandHandler`. Cette unité d'exécution extrait des messages de SYSTEM.FTE.COMMAND.*nom_agent* file d'attente de l'agent et les traite.

Si cette unité d'exécution reçoit un message contenant une demande d'agent Ping MFT, elle génère un message de réponse d'agent MFT et l'envoie à la file d'attente temporaire sur le gestionnaire de files d'attente de commandes. Ce message passe par le gestionnaire de files d'attente de l'agent.

Une fois que le message arrive dans la file d'attente temporaire, il est prélevé par la commande **ftePingAgent**. La commande écrit ensuite un message similaire à celui présenté ci-dessous sur la console avant de quitter:

```
BFGCL0213I: < nom_agent> a répondu à la commande ping en 0.088 secondes.
```

Les deux diagrammes suivants montrent le flux:

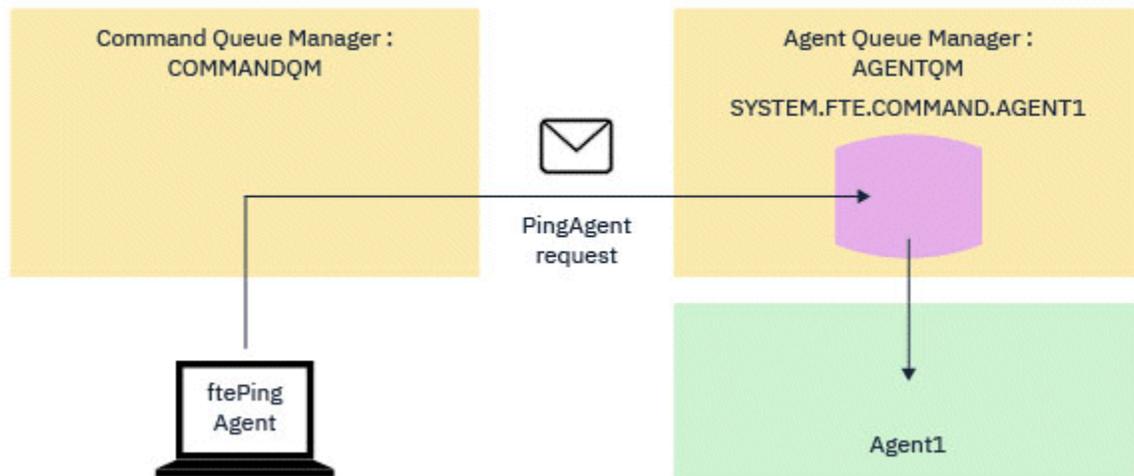


Figure 7. La demande pingAgent est envoyée à SYSTEM.FTE.COMMAND.nom_agent sur le gestionnaire de files d'attente de l'agent, via le gestionnaire de files d'attente de commandes

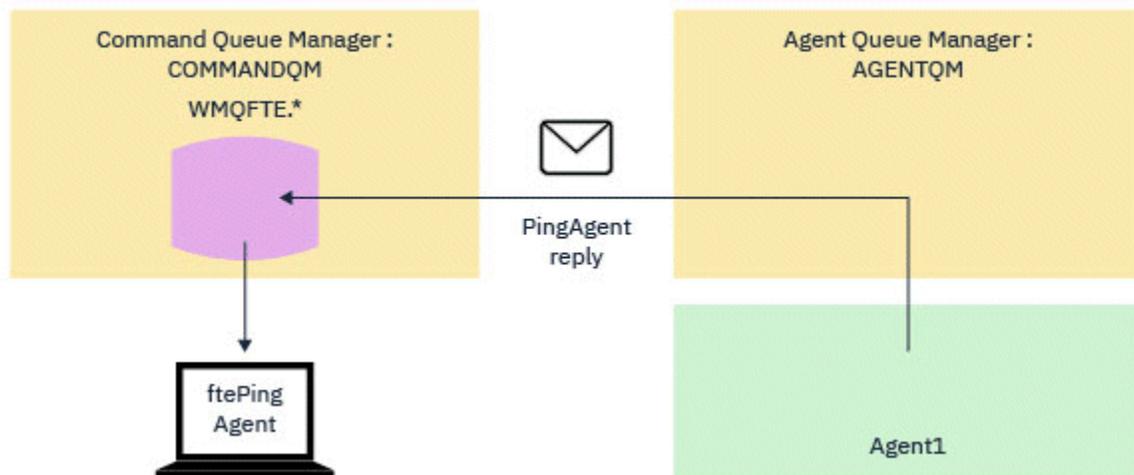


Figure 8. La réponse pingAgent revient via le gestionnaire de files d'attente de l'agent au gestionnaire de files d'attente de commandes.

Que faire si la commande dépasse le délai d'attente

Par défaut, la commande **ftePingAgent** attend pendant cinq secondes que le message de réponse de l'agent MFT arrive dans la file d'attente temporaire. Si le message de réponse n'arrive pas dans les cinq secondes, la commande écrit un message BFGCL0214I sur la console. Le message suivant est un exemple:

BFGCL0214I: agent AGENT1 n'a pas répondu à la commande ping après 5 secondes.

Vous pouvez utiliser les étapes suivantes pour déterminer la raison pour laquelle le message de réponse n'est pas arrivé:

- La première chose à faire est de vérifier que l'agent est en cours d'exécution. Si ce n'est pas le cas, il ne peut pas répondre à la demande de l'agent Ping MFT envoyée par la commande.
- Si l'agent est en cours d'exécution et qu'il est occupé à traiter des demandes, il est possible que l'obtention de la demande d'agent Ping MFT et le renvoi de la réponse prennent plus de cinq secondes.

Pour voir si c'est le cas, vous devez réexécuter la commande **ftePingAgent** en utilisant le paramètre **-w** pour spécifier un intervalle d'attente plus long. Par exemple, pour spécifier un intervalle d'attente de 60 secondes, exécutez la commande suivante:

```
ftePingAgent -w 60 AGENT1
```

- Si la commande arrive à expiration, vérifiez le chemin d'accès via le réseau IBM MQ entre le gestionnaire de files d'attente de commandes et le gestionnaire de files d'attente d'agent. Si un ou plusieurs canaux du chemin ont échoué, le message de demande de l'agent Ping MFT et/ou le message de réponse de l'agent MFT seront bloqués dans une file d'attente de transmission. Dans ce cas, vous devez redémarrer les canaux et réexécuter la commande **ftePingAgent**.

Si la commande signale toujours un message BFGCL0214I après avoir effectué les étapes précédentes, les messages de demande de l'agent Ping MFT et de réponse de l'agent MFT doivent être suivis car ils transitent par le réseau IBM MQ pour voir:

- Indique si le message de l'agent Ping MFT atteint SYSTEM.FTE.COMMAND.*nom_agent* queue.
- Si l'agent extrait le message de la file d'attente et renvoie un message de réponse de l'agent MFT.

Pour ce faire, vous devez effectuer les étapes suivantes:

- Activez les traces du gestionnaire de files d'attente sur les gestionnaires de files d'attente de commandes et d'agents.
- Activez la trace sur l'agent de manière dynamique à l'aide de la spécification de trace `com.ibm.wmqfte=all`.

La façon de procéder dépend de la plateforme sur laquelle l'agent s'exécute. Pour les agents s'exécutant sur:

- IBM MQ for Multiplatforms, voir [«Collecte dynamique d'une trace d'agent Managed File Transfer»](#), à la page 286.
- IBM MQ for z/OS, voir [«Collecte dynamique d'une trace d'agent Managed File Transfer for z/OS»](#), à la page 294.
- Ensuite, exécutez la commande **ftePingAgent** avec la fonction de trace activée, à l'aide de la spécification de trace `com.ibm.wmqfte=all`. Pour plus d'informations sur le traçage de la commande sur:
 - IBM MQ for Multiplatforms, voir [«Traçage des commandes Managed File Transfer sur Multiplatforms»](#), à la page 287.
 - IBM MQ for z/OS, voir [«Commandes Managed File Transfer for z/OS de traçage»](#), à la page 297.

Lorsque la commande arrive à expiration, arrêtez la trace de l'agent et la trace du gestionnaire de files d'attente. Les traces de l'agent et du gestionnaire de files d'attente, ainsi que la trace de la commande, doivent ensuite être mises à la disposition du support IBM pour analyse.

Que faire si votre configuration d'agent ou de consignateur MFT n'est pas sécurisée

Si un processus Managed File Transfer détecte une condition selon laquelle un fichier de configuration contient des informations sensibles, est un fichier de clés ou de clés certifiées et dispose de droits de lecture, d'écriture ou de suppression à l'échelle du système, le processus ne démarre pas s'il est détecté au moment du démarrage. Si la condition n'a pas été détectée lors du démarrage mais a été détectée lors de l'exécution, Managed File Transfer génère un message d'avertissement et ignore le contenu du fichier de configuration. Ceci est pertinent pour le pont de protocole et les capacités de pont Connect:Direct qui rechargent une configuration si elle change pendant que l'agent est en cours d'exécution.

Effectuez les vérifications suivantes pour déterminer la cause du problème:

1. Identifiez le fichier de configuration signalé comme non sécurisé à partir du message d'erreur fourni.

2. Vérifiez que les droits d'accès aux fichiers correspondent aux exigences requises. Pour plus d'informations, voir la section [«Droits MFT d'accès aux informations de configuration sensibles»](#), à la page 372.
3. Redémarrez l'agent ou le consignateur. Ou, dans le cas du pont de protocole ou des fichiers de données d'identification Connect:Direct , attendez le prochain rechargement.

Exemple

Dans cet exemple de message d'erreur, un consignateur de base de données ne démarre pas:

```
BFGDB0066E: The logger encountered a problem accessing its credentials file and will stop.  
Reported error: BFGNV0145E: The 'Everyone' group has access to the file 'C:\mqmftcredentials.xml'.
```

Dans cet exemple de message d'erreur, un agent de pont de protocole ne démarre pas:

```
BFGI00383E: The security permissions defined for credentials file 'C:\ProtocolBridgeCredentials.xml' do  
not meet the  
minimum requirements for a file of this type.  
Reported problem: BFGNV0145E: The 'Everyone' group has access to the file  
C:\ProtocolBridgeCredentials.xml'.
```

Référence associée

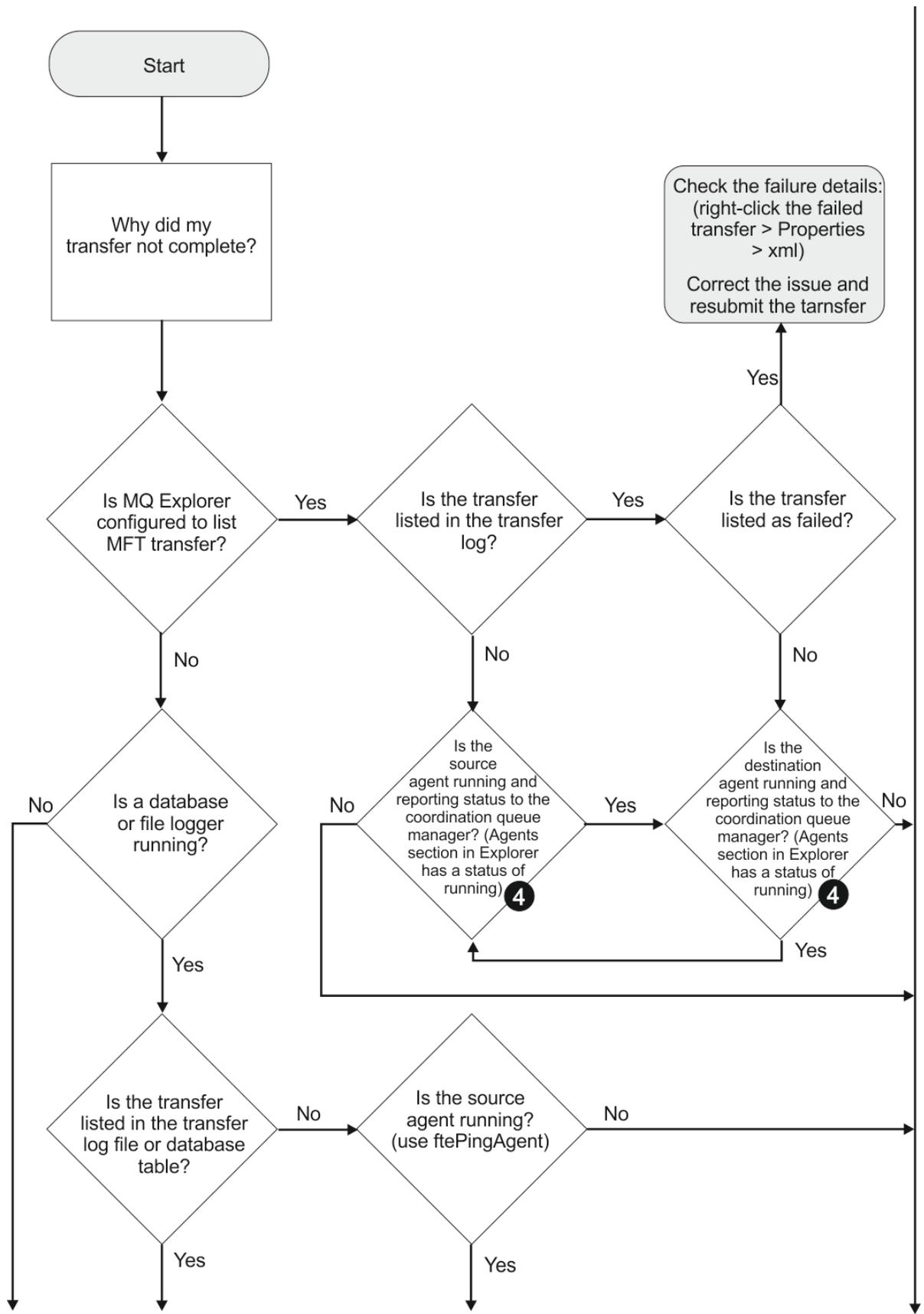
[«Droits MFT d'accès aux informations de configuration sensibles»](#), à la page 372

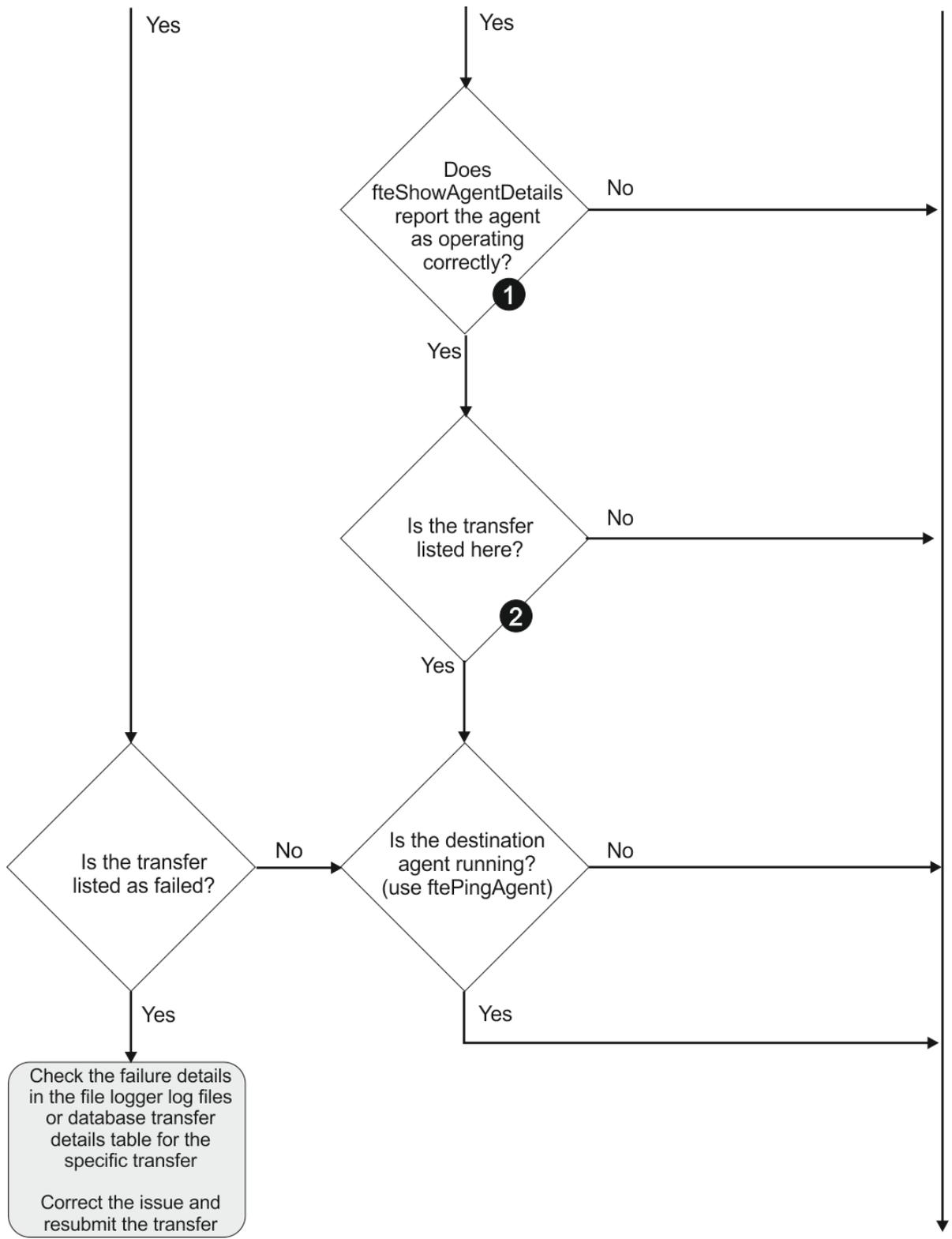
Tout fichier utilisé pour stocker des informations de configuration sensibles, c'est-à-dire tout fichier référencé à partir de l'arborescence de configuration IBM MQ , ne doit pas disposer de droits de lecture, d'écriture ou (le cas échéant) de suppression à l'échelle du système. Ces restrictions s'appliquent également aux fichiers de clés certifiées et aux fichiers de clés.

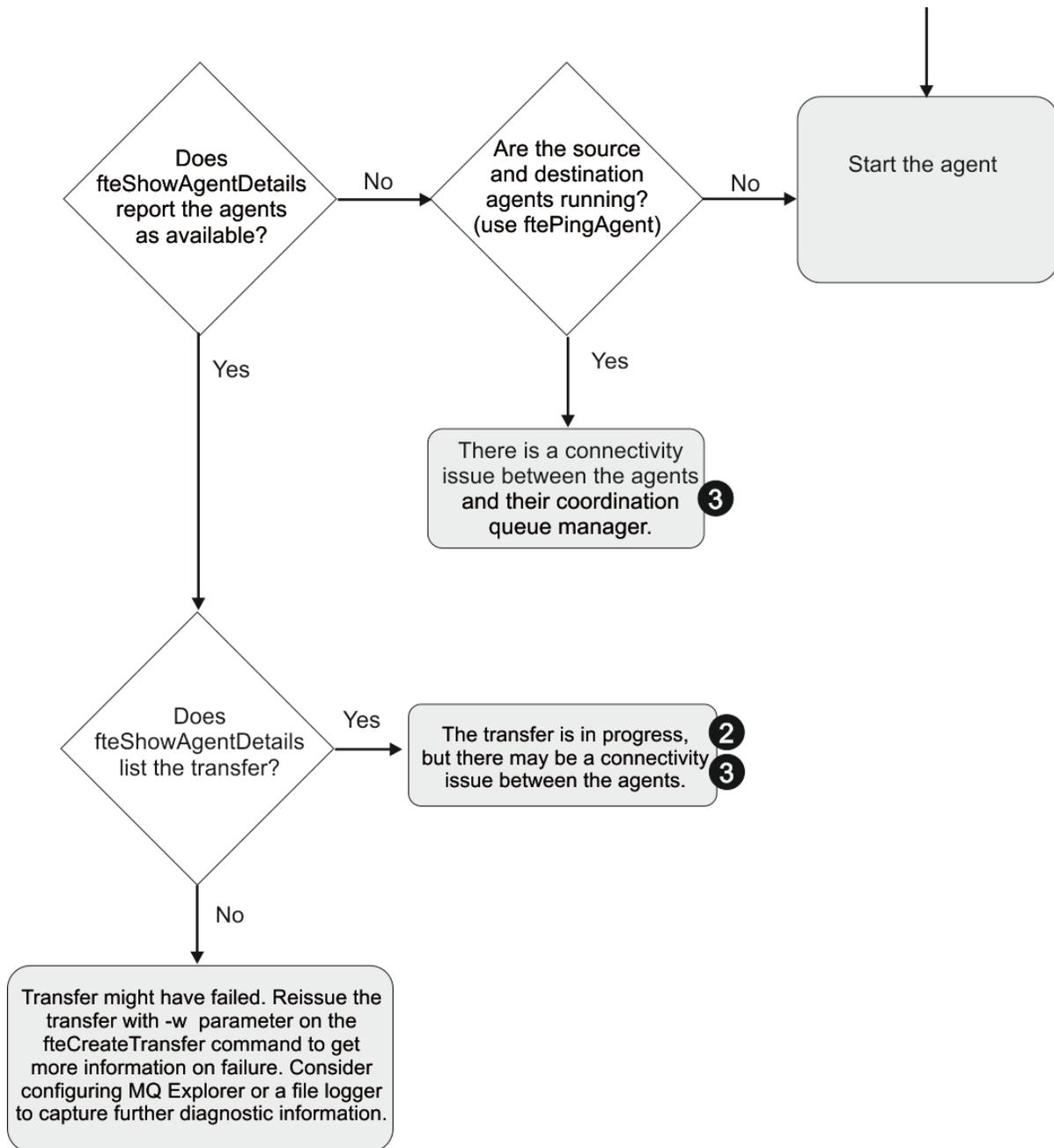
Que faire si votre transfert n'est pas terminé

Si votre transfert n'est pas terminé, vous pouvez effectuer un certain nombre d'étapes d'identification de problème pour en déterminer la cause.

Servez-vous du diagramme suivant pour diagnostiquer les problèmes et décider de l'action à effectuer :







Clé du diagramme :

1. Vérifiez si l'agent `output0.log` contient des erreurs. Si l'agent indique qu'il a démarré correctement, mais que ni IBM MQ Explorer ni **fteShowAgentDetails** ne signalent que l'agent est en cours d'exécution, vérifiez la connectivité entre le gestionnaire de files d'attente de l'agent et le gestionnaire de files d'attente de coordination. Il se peut qu'un gestionnaire de files d'attente vers un canal de gestionnaire de files d'attente ne soit pas disponible.
2. Si l'agent source répertorie l'ID de transfert en tant que transfert In progress mais pas l'agent de destination, il se peut qu'il y ait un problème de connectivité entre les gestionnaires de files d'attente source et de destination. Utilisez la commande **ftePingAgent** de la machine de l'agent cible vers l'agent source en utilisant le gestionnaire de files d'attente de l'agent cible comme gestionnaire de files d'attente de commandes, dans le fichier `command.properties`. Vous pouvez également exécuter cette commande dans l'autre sens, de la source à la destination.
3. Si les agents source et de destination répertorient l'ID de transfert sous la forme In progress, cela indique qu'il y a eu un problème de connectivité entre les gestionnaires de files d'attente source et

de destination depuis le début du transfert. Utilisez la commande **ftePingAgent** de la machine de l'agent cible vers l'agent source en utilisant le gestionnaire de files d'attente de l'agent cible comme gestionnaire de files d'attente de commandes, dans le fichier `command.properties`. Vous pouvez également exécuter cette commande dans l'autre sens, de la source à la destination.

4. Si vous avez déjà fait le tour de cette boucle, vérifiez si l'une ou l'autre des instructions est pertinente pour votre situation:
 - Les agents source et de destination indiquent `Running`, mais aucun transfert n'est répertorié. Soit la demande de transfert n'a pas atteint la file d'attente de commandes de l'agent, soit l'agent, bien qu'il se soit signalé comme `Running`, ne surveille plus la file d'attente de commandes. Recherchez les erreurs dans l'agent source `output0.log`. Utilisez la commande **ftePingAgent** à partir de la même machine à partir de laquelle le transfert a été envoyé à l'agent source, pour vérifier la connectivité entre le gestionnaire de files d'attente de commandes et le gestionnaire de files d'attente d'agents, et que l'agent gère la file d'attente de commandes.
 - Les agents source et de destination indiquent `Running` et le transfert est répertorié en tant que `In progress, recovering`. Utilisez la commande **ftePingAgent** de la machine de l'agent cible vers l'agent source en utilisant le gestionnaire de files d'attente de l'agent cible comme gestionnaire de files d'attente de commandes, dans le fichier `command.properties`. Vous pouvez également exécuter cette commande dans l'autre sens, de la source à la destination.

Que faire si vous pensez que votre transfert de fichiers est bloqué

Sur un système très chargé ou en cas de problèmes de réseau entre les agents source et cible, les transferts peuvent parfois sembler bloqués dans un état de file d'attente ou de récupération. Un certain nombre de facteurs peuvent être à l'origine de cette situation.

Effectuez les vérifications suivantes pour déterminer la cause du problème:

1. Utilisez la commande **ftePingAgent** ou, dans le panneau IBM MQ Explorer **Agents**, cliquez avec le bouton droit de la souris sur le nom de l'agent et sélectionnez **Ping** pour vérifier si les agents source et de destination sont actifs et répondent aux nouvelles demandes. Consultez les journaux de l'agent pour voir s'il existe un problème de connexion réseau en cours.
2. Vérifiez si l'agent de destination est en cours d'exécution à pleine capacité. Il se peut qu'il existe de nombreux agents source qui demandent tous des transferts de fichiers vers le même agent cible. Utilisez la commande **fteShowAgentDetails** avec le paramètre **-v** (prolix) ou dans le panneau IBM MQ Explorer **Agents**, cliquez avec le bouton droit de la souris sur le nom de l'agent et sélectionnez **Propriétés** pour afficher l'activité de transfert en cours pour un agent. Si le nombre de transferts de destination en cours d'exécution est égal ou proche du nombre maximal de transferts de destination de l'agent, cela peut expliquer pourquoi certains transferts pour les agents source semblent être bloqués.
3. Les transferts vers et depuis les agents de pont de protocole passent à l'état de récupération en cas de problème lors de la prise de contact avec le serveur de fichiers de protocole. Consultez les journaux de l'agent pour voir s'il existe un problème de connexion en cours.
4. Les transferts sont traités par un agent dans l'ordre de priorité. Par conséquent, dans un système chargé, un transfert de priorité basse peut rester en file d'attente pendant un certain temps pendant que l'agent est chargé avec des transferts de priorité plus élevée. Finalement, un transfert de priorité basse est démarré si ce transfert a été mis en file d'attente pendant un certain temps, même s'il existe de nouveaux transferts de priorité plus élevée.

Que faire si votre transfert de fichiers planifié ne s'exécute pas ou est retardé

Si vous disposez d'un transfert planifié qui ne s'exécute pas lorsqu'il est dû ou retardé, il se peut que l'agent traite des commandes dans sa file d'attente de commandes. L'agent étant occupé, les transferts planifiés ne sont pas vérifiés et ne sont donc pas exécutés.

Pour contourner ce problème, utilisez l'une des étapes suivantes:

- Configurez la propriété `maxSchedulerRunDelay` dans le fichier `agent.properties` pour définir l'intervalle maximal en minutes pendant lequel l'agent attend de vérifier les transferts planifiés. La définition de cette propriété garantit que l'agent continue de vérifier les transferts planifiés même lorsque l'agent est occupé. Pour plus d'informations sur cette propriété, voir [«Le fichier MFT agent.properties»](#), à la page 564.
- Vous pouvez également utiliser un moniteur de ressources à la place d'un transfert planifié. Les moniteurs de ressources fonctionnent différemment des transferts planifiés et ne sont pas affectés par l'occupation de l'agent. Par exemple, si vous souhaitez un fichier à jour sur le système de destination, les moniteurs de ressources réduisent le trafic réseau. Cela est dû au fait que le fichier est transféré uniquement lorsqu'une nouvelle version devient disponible, plutôt que le fichier est transféré automatiquement. Toutefois, la surveillance des ressources n'est pas prise en charge sur les agents de pont de protocole ou les agents de pont Connect:Direct .

Pour plus d'informations, voir la section [«Surveillance des ressources MFT»](#), à la page 174.

Que faire si votre agent de pont de protocole signale qu'un fichier est introuvable

Lorsque l'agent de pont de protocole signale que le serveur SFTP ou FTP auquel se connecte le pont de protocole renvoie un message d'erreur `File not found`, ce message peut indiquer que l'un des nombreux cas d'erreur s'est produit.

Les scénarios suivants peuvent entraîner le renvoi d'une erreur `File not found` par le serveur SFTP ou FTP.

- Le fichier n'existe pas. Vérifiez que le fichier que vous tentez de transférer existe sur le système hébergeant le serveur SFTP ou FTP.
- Le chemin d'accès au fichier n'existe pas. Vérifiez que le chemin d'accès au fichier existe sur le système hébergeant le serveur SFTP ou FTP. Vérifiez que vous avez entré correctement le chemin d'accès au fichier dans la demande de transfert. Si nécessaire, corrigez le chemin d'accès au fichier et soumettez à nouveau la demande de transfert.
- Le fichier est verrouillé par une autre application. Vérifiez si le fichier est verrouillé par une autre application. Attendez que le fichier ne soit plus verrouillé, puis soumettez à nouveau la demande de transfert.
- Les droits d'accès au fichier n'autorisent pas la lecture du fichier. Vérifiez si le fichier possède les droits d'accès appropriés. Si nécessaire, modifiez les droits d'accès aux fichiers et soumettez à nouveau la demande de transfert.
- Le serveur SFTP ou FTP utilise un chemin racine virtualisé. Si un chemin de fichier relatif est spécifié dans une demande de transfert, l'agent de pont de protocole tente de convertir le chemin relatif en chemin de fichier absolu en fonction du répertoire de base utilisé pour la connexion au serveur. The Managed File Transfer ne peut prendre en charge que les serveurs SFTP ou FTP qui permettent l'accès aux fichiers par leur chemin de fichier absolu. Les serveurs de protocole qui autorisent l'accès aux fichiers basés uniquement sur le répertoire en cours ne sont pas pris en charge par l'agent de pont de protocole.

Concepts associés

[«Le pont de protocole»](#), à la page 219

Le pont de protocole permet à votre réseau Managed File Transfer (MFT) d'accéder aux fichiers stockés sur un serveur de fichiers en dehors de votre réseau MFT, dans votre domaine local ou à un emplacement distant. Ce serveur de fichiers peut utiliser les protocoles de réseau FTP, FTPS ou SFTP. Chaque serveur de fichiers a besoin d'au moins un agent dédié. L'agent dédié est appelé agent de pont de protocole. Un agent de pont peut interagir avec plusieurs serveurs de fichiers.

Que faire si votre moniteur de ressources de répertoire MFT ne déclenche pas de fichiers

Un moniteur de ressources de répertoire recherche dans un répertoire les fichiers qui correspondent à une spécification de déclencheur. Pour chaque fichier correspondant à la spécification du déclencheur,

une demande de transfert est générée pour l'agent. Lorsque la demande est soumise, le fichier de déclenchement est ignoré jusqu'à ce que le fichier soit modifié.

Raisons possibles pour lesquelles les fichiers ne sont pas déclenchés

1. Le moniteur de ressources de répertoire a trouvé un fichier correspondant à la spécification de déclencheur, mais la demande de transfert générée n'était pas valide et l'agent n'a pas pu traiter la demande. Les raisons peuvent être les suivantes:

- Agent de destination non valide
- Agent de destination manquant
- Transfert annulé par appel de programme

Dans tous ces exemples, le moniteur de ressources de répertoire marque le fichier de déclenchement comme traité et ignore le fichier même si le transfert a échoué.

2. Le fichier est en dehors de la portée de la spécification de déclencheur du moniteur de ressources. Les raisons peuvent être les suivantes:

- Modèle de déclencheur incorrect
- Surveillance du répertoire incorrect
- Droits d'accès aux fichiers insuffisants
- Echec de la connexion au système de fichiers distant

Pourquoi un fichier peut déclencher un second transfert

Un fichier de déclenchement peut générer une demande de transfert Managed File Transfer pour les raisons suivantes:

- Si la présence du fichier de déclenchement est détectée, alors qu'il n'y était pas auparavant.
- Si le fichier de déclenchement a été mis à jour, la date de dernière modification est modifiée.

Les scénarios possibles pour un deuxième déclencheur sont les suivants:

- Le fichier est supprimé, puis remplacé.
- Le fichier est verrouillé par une application, puis déverrouillé par une autre application.
- Le système de fichiers du moniteur échoue. Par exemple, si la connexion réseau échoue, cela peut donner l'apparence du fichier supprimé, puis remplacé.
- Le répertoire de fichiers est mis à jour par une autre application, ce qui entraîne le changement de la date de dernière modification.

Commande permettant de définir la sortie de niveau d'information pour tous les moniteurs de ressources d'un agent

V 9.0.3

Dans cet exemple, tous les moniteurs de ressources sont surveillés car vous n'avez pas spécifié de nom, ou de nom, pour des moniteurs de ressources spécifiques. Le nom de l'agent est AGENT1.

```
fteSetAgentLogLevel -logMonitor=info AGENT1
```

Pour plus d'informations sur le paramètre **logMonitor** et des exemples d'utilisation des différentes options, voir [fteSetAgentLogLevel](#).

Exemple de sortie de niveau d'information pour tous les moniteurs de ressources d'un agent

V 9.0.3

```
=====
```

```
[21/04/2017 11:08:49:367 IST] BFGUT0036I: Resource monitor event log level has changed to "info" for all resource monitors of this agent.
```

```
=====
Date          Time          Thread ID   Monitor Name   Event
Description
=====
[21/04/2017 11:08:51:842 IST] 00000023   QMON           Monitor Started   Resource Monitor Started
[21/04/2017 11:08:51:844 IST] 00000025   QMON           Start Poll        New poll cycle started
[21/04/2017 11:08:51:924 IST] 00000023   MON1          Monitor Started   Resource Monitor Started
[21/04/2017 11:08:51:925 IST] 00000026   MON1          Start Poll        New poll cycle started
[21/04/2017 11:08:52:029 IST] 00000026   MON1          End Poll          Poll cycle completed in 105
milli seconds. Trigger items matched [ 0 ]
[21/04/2017 11:08:52:055 IST] 00000025   QMON           End Poll          Poll cycle completed in 212
milli seconds. Trigger items matched [ 0 ]
[21/04/2017 11:09:51:840 IST] 00000025   QMON           Start Poll        New poll cycle started
[21/04/2017 11:09:51:875 IST] 00000025   QMON           End Poll          Poll cycle completed in 34
milli seconds. Trigger items matched [ 0 ]
[21/04/2017 11:09:51:924 IST] 00000026   MON1          Start Poll        New poll cycle started
[21/04/2017 11:09:51:969 IST] 00000026   MON1          End Poll          Poll cycle completed in 45
milli seconds. Trigger items matched [ 0 ]
[21/04/2017 11:10:51:840 IST] 00000025   QMON           Start Poll        New poll cycle started
[21/04/2017 11:10:51:924 IST] 00000026   MON1          Start Poll        New poll cycle started
[21/04/2017 11:10:51:962 IST] 00000025   QMON           End Poll          Poll cycle completed in 121
milli seconds. Trigger items matched [ 0 ]
[21/04/2017 11:10:51:963 IST] 00000026   MON1          End Poll          Poll cycle completed in 39
milli seconds. Trigger items matched [ 0 ]
[21/04/2017 11:10:55:063 IST] 00000041   MON1          Monitor Stopped   Resource Monitor Stopped
[21/04/2017 11:10:55:079 IST] 00000041   QMON          Monitor Stopped   Resource Monitor Stopped
=====
```

Référence associée

[«fteSetAgentLogLevel: Définir le niveau de journalisation de l'agent de pont de protocole MFT», à la page 512](#)

La commande **fteSetAgentLogLevel** permet d'activer ou de désactiver la consignation des informations de diagnostic pour les transferts de fichiers entre un agent de pont de protocole Managed File Transfer et des serveurs de fichiers FTP/SFTP/FTPS.

Que faire si les fichiers de destination créés par un transfert démarré par un moniteur de ressources de file d'attente contiennent des données incorrectes

Vous pouvez créer un moniteur de ressources pour surveiller une file d'attente et transférer un message ou un groupe de messages d'une file d'attente vers un fichier. Le nom de fichier peut être spécifié à l'aide des descripteurs de message MQMD du message ou du premier message d'un groupe. Si un transfert de message à fichier échoue et que le message ou le groupe est laissé dans la file d'attente, la prochaine fois que le moniteur est déclenché, il peut entraîner la création de fichiers contenant des données incorrectes.

Pourquoi ce problème se produit

1. Un transfert de message à fichier échoue et le message ou le groupe reste dans la file d'attente.
2. Un nouveau message ou groupe arrive dans la file d'attente.
3. Le nouveau message ou groupe déclenche le moniteur de ressources.
4. Le moniteur de ressources crée un nouveau transfert qui utilise les descripteurs de message MQMD du nouveau message ou groupe et les données du premier message ou groupe de la file d'attente.
5. Des fichiers contenant des données incorrectes sont créés.

Eviter ce problème

Pour éviter ce problème, vous devez créer manuellement un fichier de définition de transfert à l'aide de la commande **fteCreateTransfer** et éditer l'élément <queue> du fichier pour inclure l'attribut `groupId="{GROUPID}"`. Soumettez ensuite le fichier de définition de transfert à l'aide de la commande **fteCreateMonitor**.

Exemple

Dans cet exemple, l'agent source, qui est également l'agent de surveillance, est appelé AGENT_MON ; l'agent de destination est appelé AGENT_DEST ; le nom du fichier de destination est /out/files/{WMQFTEFileName}. Cet exemple requiert que le descripteur de message MQMD WMQFTEFileName soit défini pour le message. La file d'attente surveillée est LIVE_QUEUE.

1. Créez un fichier de définition de transfert en exécutant la commande suivante:

```
fteCreateTransfer -sa AGENT_MON -da AGENT_DEST -df "/out/files/${WMQFTEFileName}"  
-de error -gt /tmp/TransferDefinition1.xml -sqgi -sq LIVE_QUEUE
```

Le fichier de définition de transfert /tmp/TransferDefinition1.xml est généré.

2. Modifiez l'élément <queue> pour inclure l'attribut groupId=" \${GROUPID} ". Remplacez la ligne

```
<queue useGroups="true">LIVE_QUEUE</queue>
```

à

```
<queue useGroups="true" groupId="${GROUPID}">LIVE_QUEUE</queue>
```

Cet attribut est requis pour que le transfert lise le groupe ou le message qui a déclenché le transfert à partir de la file d'attente à la place du premier groupe ou message de la file d'attente.

3. Créez le moniteur en exécutant la commande suivante:

```
fteCreateMonitor -ma AGENT_MON -mq LIVE_QUEUE -mn QueueMon1 -mt /tmp/TransferDefinition1.xml  
-tr completeGroups -dv WMQFTEFileName=UNKNOWN
```

Ce moniteur interroge la file d'attente toutes les 60 secondes pour voir si un nouveau groupe ou message est arrivé dans la file d'attente.

Que faire si la file d'attente de destination est une file d'attente en cluster ou un alias d'une file d'attente en cluster

Lorsque vous utilisez Managed File Transfer pour transférer un fichier dans une file d'attente, si vous utilisez une destination qui est une file d'attente en cluster ou un alias dans une file d'attente en cluster, vous obtenez le code raison 2085 ou 2082. A partir de IBM WebSphere MQ 7.5.0 Fix Pack 4 , ce problème est résolu si vous affectez la valeur true à la propriété enableClusterQueueInputOutput.

Pourquoi ce problème se produit

Le nom du gestionnaire de files d'attente de l'agent de destination est ajouté au nom de file d'attente du paramètre **-dq** lorsqu'il n'existe pas de nom de gestionnaire de files d'attente explicite sur le **-dq**. Le code anomalie 2085 ou 2082 est généré car l'objet queueManager ne peut pas être spécifié sur un appel MQOPEN lors de la connexion à une file d'attente en cluster MQ queueManager qui ne comporte pas cette file d'attente en cluster locale.

Eviter ce problème

1. Créez une file d'attente en cluster sur le gestionnaire de files d'attente.
2. Configurez une définition de file d'attente éloignée qui pointe vers une file d'attente en cluster.

Exemple

Cet exemple utilise une définition de file d'attente éloignée.

Configuration :

- Agent source: *SAGENT*
- Gestionnaire de files d'attente de l'agent source: *SQM*
- Agent de destination: *DAGENT*
- Gestionnaire de files d'attente de l'agent de destination: *DQM*
- La file d'attente de destination du transfert est *CQ6* sur le gestionnaire de files d'attente *SQM*

Pour définir la définition de file d'attente éloignée Q6_SQM sur DQM dans la file d'attente en cluster CQ6 dans SQM (en supposant que la file d'attente en cluster CQ6 est déjà définie dans SQM), émettez la commande MQSC sur le gestionnaire de files d'attente DQM:

```
define qremote(Q6_SQM) rname(CQ6) rqnname(SQM) xmitq(SQM)
```

Remarque : rname pointe vers la file d'attente en cluster.

Vous pouvez maintenant effectuer un transfert vers la file d'attente. Exemple :

```
fteCreateTransfer -sa SAGENT -sm SQM -da DAGENT -dm DQM -dq Q6_SQM /tmp/single_record.txt
```

Que faire si des messages sont générés dans votre système SYSTEM.MANAGED.DURABLE files d'attente ou remplissage de votre système de fichiers

Si votre plug-in IBM MQ Explorer utilise un abonnement durable sur le gestionnaire de files d'attente de coordination, les messages peuvent s'accumuler dans 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.

Permet de supprimer l'accumulation de messages sur le système SYSTEM.MANAGED.DURABLE , vous pouvez effectuer l'une des actions suivantes:

- Démarrez le IBM MQ Explorer qui utilise l'abonnement durable. Le plug-in Managed File Transfer pour IBM MQ Explorer consomme les messages de la file d'attente.
- Supprimez manuellement les messages des files d'attente.

V 9.0.2

Vous pouvez éviter la génération de messages dans des files d'attente durables de l'une des manières suivantes:

- 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 **Fenêtre > Préférences > IBM MQ Explorer > Managed File Transfer**
 2. Dans la liste **Type d'abonnement au journal de transfert**, choisissez NON_DURABLE.
- Effacez les abonnements durables du gestionnaire de files d'attente de coordination créés par le plug-in IBM MQ Explorer MFT .

Le nom de l'abonnement durable est préfixé pour indiquer que l'abonnement a été créé avec le plug-in IBM MQ Explorer MFT, le nom d'hôte et le nom de l'utilisateur, par exemple MQExplorer_MFT_Plugin_HOST_TJWatson.

Référence associée

«Conservation des messages de journal MFT», à la page 106

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.

Examen des messages avant la publication

Etant donné que les agents peuvent se connecter aux gestionnaires de files d'attente IBM WebSphere MQ 6.0 , ils n'utilisent pas l'approche de publication directe introduite dans IBM WebSphere MQ 7.0. A la place, les agents envoient des messages ordinaires au gestionnaire de files d'attente de coordination qui contient un en-tête MQRFH. L'en-tête MQRFH demande que le contenu du message soit publié. Ces messages sont envoyés à SYSTEM.FTE sur le gestionnaire de file d'attente de coordination et les

messages sont généralement publiés immédiatement à partir de cette file d'attente. Si des conditions d'erreur arrêtent cette publication, vous pouvez examiner les messages de la file d'attente avant de tenter de la publier pour faciliter le diagnostic. Pour ce faire, procédez comme suit:

1. Désactivez le moteur de publication / abonnement dans le gestionnaire de files d'attente de coordination.

Vous pouvez effectuer cette étape à l'aide de IBM MQ Explorer ou des commandes MQSC. Sachez que cela arrête temporairement toute activité de publication / abonnement sur le gestionnaire de files d'attente, y compris toute activité non liée à Managed File Transfer si votre gestionnaire de files d'attente de coordination est également utilisé à d'autres fins.

IBM MQ Explorer:

- a. Dans la vue Navigator , cliquez avec le bouton droit de la souris sur le gestionnaire de files d'attente de coordination et sélectionnez **Propriétés**.
- b. Dans la sous-fenêtre **Propriétés** , sélectionnez **Publier / S'abonner**.
- c. Sélectionnez **Compatibility** dans la liste **Publish / Subscribe mode** .

mqsc:

```
ALTER QMGR PSMODE(COMPAT)
```

2. Envoyez un autre message.

Effectuez l'action Managed File Transfer qui présente des problèmes de publication. Par exemple, pour l'enregistrement d'agent, un message est envoyé chaque fois que l'agent est démarré (vous n'avez pas besoin de supprimer et de créer l'agent pour générer des messages d'enregistrement). Le moteur de publication / abonnement étant désactivé, aucune publication n'a lieu.

3. Parcourez SYSTEM.FTE FTE sur le gestionnaire de file d'attente de coordination.

Vous devez utiliser IBM MQ Explorer pour parcourir le système SYSTEM.FTE , file d'attente.

IBM MQ Explorer:

- a. Dans la vue Navigator , développez le gestionnaire de files d'attente de coordination et cliquez sur **Files d'attente**. Dans la vue Contenu, cliquez avec le bouton droit de la souris sur SYSTEM.FTE , sélectionnez **Parcourir les messages**. La fenêtre **Message browser** s'ouvre et affiche les messages qui auraient été publiés.
- b. La colonne **ID utilisateur** affiche l'ID utilisateur contenu dans le descripteur de message. L'une des raisons courantes de l'échec de la publication est que cet ID utilisateur ne dispose pas de l'autorisation de publication sur SYSTEM.FTE .
- c. Pour plus d'informations sur chaque message (y compris le XML qui sera publié), cliquez avec le bouton droit de la souris sur le message et sélectionnez **Propriétés**.

Il n'existe aucune commande MQSC permettant d'inspecter le contenu des messages. Si vous ne disposez pas du IBM MQ Explorer, vous devez utiliser un autre programme qui peut parcourir les files d'attente et afficher tous les aspects des messages trouvés. Vous pouvez utiliser l'exemple de programme **amqsbcg** , s'il est installé, comme décrit dans la rubrique suivante: [Files d'attente de navigation](#). La ligne `UserIdentifier` affiche l'ID utilisateur. Vous pouvez également utiliser **dmpmqmsg**; l'ID utilisateur d'un message se trouve dans des lignes telles que:

```
A RTM MQ24
A USR JOHND0E
A ACC 1A0FD4D8F2F4C3C8C9D5F1F9C6F7C1C3F3F00019F7AC3000000000000000000
```

La deuxième ligne de l'exemple est l'ID utilisateur du descripteur de message pour ce message.

4. Réactivez le moteur de publication / abonnement du gestionnaire de files d'attente de coordination.

Vous pouvez effectuer cette étape à l'aide de IBM MQ Explorer ou des commandes MQSC. Une fois que vous avez réactivé le moteur de publication / abonnement dans le gestionnaire de files d'attente de coordination, tous les messages dans SYSTEM.FTE est traitée immédiatement.

IBM MQ Explorer:

- Dans la vue Navigator , cliquez avec le bouton droit de la souris sur le gestionnaire de files d'attente de coordination et sélectionnez **Propriétés**.
- Dans la sous-fenêtre **Propriétés** , sélectionnez **Publier / S'abonner**.
- Sélectionnez **Activé** dans la liste **Mode de publication / abonnement** .

mqsc:

```
ALTER QMGR PSMODE(ENABLED)
```

IBM i Erreurs possibles lors du transfert des fichiers sauvegarde IBM i

Si vous utilisez Managed File Transfer pour transférer plusieurs fois le même fichier de sauvegarde IBM i , le transfert risque d'échouer.

Managed File Transfer peut générer l'une des erreurs suivantes ou les deux:

- BFGII0003E: Unable to open file "/qsys.lib/library.lib/SAVF.FILE" for reading
- BFGII0082E: A file open for read failed due to a Java IOException with message text "Sharing violation occurred"

Ces erreurs peuvent se produire si vous émettez plusieurs demandes simultanées pour qu'un agent MFT transfère le même fichier sauvegarde IBM i . Si vous souhaitez transférer simultanément le même fichier sauvegarde plusieurs fois, vous devez utiliser plusieurs agents source. Utilisez un agent source différent pour chaque transfert simultané.

Pour transférer le même fichier sauvegarde plusieurs fois avec un seul agent source, vous devez attendre que la demande de transfert précédente soit terminée avant de soumettre chaque nouvelle demande de transfert.

V 9.0.1 Erreurs possibles lors de la configuration de l'agent MFT redistribuable

Messages d'erreur lors de la configuration du Redistributable Managed File Transfer Agent

La bibliothèque native pour Windows n'a pas pu être chargée

Windows Vous devez installer les bibliothèques Microsoft suivantes sur votre système pour utiliser Redistributable Managed File Transfer Agent:

- Microsoft Visual C++ Redistributable 2008
- Microsoft Visual C++ Redistributable 2012

Ces bibliothèques sont disponibles dans Microsoft. Voir [Les derniers téléchargements Visual C++ pris en charge](#).

Si ces bibliothèques ne sont pas installées et que vous tentez d'exécuter des commandes MFT , une erreur est signalée:

- BFGUB0070E: Internal error: Native library for platform Windows 7 (architecture amd64) could not be loaded because mqmft (Not found in java.library.path).

- BFGCL0043I: Specify the '-h' command line parameter to see more usage information.

Vérifiez que les bibliothèques Microsoft sont installées. Si les bibliothèques ne sont pas installées, installez-les et relancez la commande.

L'utilisation du mode liaisons n'est pas prise en charge

Redistributable Managed File Transfer Agent peut uniquement se connecter à IBM MQ en mode client. Si vous essayez d'exécuter des commandes en mode liaisons, une erreur est signalée:

- BFGCL0408E: Unable to obtain IBM MQ installation information for queue manager '*queue manager name*'. Reason Cannot run program "../bin/dspmq": error=2, No such file or directory

Lorsque vous émettez des commandes, vous devez indiquer l'hôte, le port, le nom et le nom du canal du gestionnaire de files d'attente.

Conseils pour la définition des attributs MQ et des propriétés MFT associées à la taille des messages

Vous pouvez modifier les attributs IBM MQ et les propriétés Managed File Transfer pour affecter le comportement de Managed File Transfer lors de la lecture ou de l'écriture de messages de différentes tailles.

Si la taille des messages lus à partir d'une file d'attente source ou écrits dans une file d'attente de destination dépasse 1048576 octets (1 Mo), vous devez augmenter la valeur de la propriété Managed File Transfer Agent **maxInputOutputMessageLength** à une valeur supérieure ou égale à la taille maximale des messages à lire ou à écrire.

Si les messages de la file d'attente source sont supérieurs à 1048576 octets, vous devez définir la propriété **maxInputOutputMessageLength** sur l'agent source. Si les messages de la file d'attente de destination sont supérieurs à 1048576 octets, vous devez définir la propriété **maxInputOutputMessageLength** sur l'agent de destination. Pour plus d'informations sur la propriété **maxInputOutputMessageLength**, voir [Propriétés avancées de l'agent](#).

- Si la file d'attente dans laquelle l'agent est en cours d'écriture ou de lecture est locale dans le gestionnaire de files d'attente de l'agent, vous devrez peut-être modifier les attributs du IBM MQ gestionnaire de files d'attente, de la file d'attente et du canal **MAXMSGL**.

Vérifiez que la valeur de la taille de message maximale de la file d'attente source ou de destination est supérieure ou égale à la valeur de la propriété d'agent **maxInputOutputMessageLength**.

Vérifiez que la valeur de chacun des attributs IBM MQ suivants, en octets:

- Taille maximale des messages du gestionnaire de files d'attente de l'agent
- Taille maximale des messages du système SYSTEM.FTE.STATE.*nom_agent* file d'attente
- Taille maximale des messages du canal client, si votre agent se connecte au gestionnaire de files d'attente en mode client

est supérieur ou égal au résultat du calcul suivant:

Pour un transfert de fichier à message (qui prend en charge une taille de fichier pouvant atteindre 100 Mo):

La valeur de **maxInputOutputMessageLength**

Pour un transfert de message vers fichier:

La valeur de $3 * (\text{maxInputOutputMessageLength}) + 1048576$

(Ce calcul est dérivé du fait que trois points de contrôle peuvent être stockés dans un message d'état et que chaque point de contrôle peut avoir à mettre en mémoire tampon jusqu'à la taille maximale d'une quantité de données de message.)

- Si la file d'attente dans laquelle l'agent écrit est une file d'attente éloignée, vous devrez peut-être modifier les attributs **MAXMSGL** du gestionnaire de files d'attente, de la file d'attente et du canal IBM MQ.

Vérifiez que la valeur de chacun des attributs IBM MQ suivants est supérieure ou égale à la valeur de la propriété d'agent **maxInputOutputMessageLength** :

- Taille maximale des messages de la file d'attente de transmission du gestionnaire de files d'attente éloignées sur le gestionnaire de files d'attente de l'agent
- Taille maximale des messages du canal entre le gestionnaire de files d'attente de l'agent et le gestionnaire de files d'attente éloignées
- Taille maximale des messages de la file d'attente de destination sur le gestionnaire de files d'attente éloignées
- Taille maximale des messages du gestionnaire de files d'attente éloignées

Vérifiez que la valeur de chacun des attributs IBM MQ suivants, en octets:

- Taille maximale des messages du gestionnaire de files d'attente de l'agent
- Taille maximale des messages du système SYSTEM.FTE.STATE.*nom_agent* file d'attente
- Taille maximale des messages du canal client, si votre agent se connecte au gestionnaire de files d'attente en mode client

est supérieur ou égal au résultat du calcul suivant:

Pour un transfert de fichier à message (qui prend en charge une taille de fichier pouvant atteindre 100 Mo):

La valeur de **maxInputOutputMessageLength**

Pour un transfert de message vers fichier:

La valeur de $3 * (\text{maxInputOutputMessageLength}) + 1048576$

(Ce calcul est dérivé du fait que trois points de contrôle peuvent être stockés dans un message d'état et que chaque point de contrôle peut avoir à mettre en mémoire tampon jusqu'à la taille maximale d'une quantité de données de message.)

Si vous dépassez la valeur de l'une de ces propriétés, l'agent s'arrête avec l'erreur suivante dans le journal des événements de l'agent:

```
BFGUT0002E: An internal error has occurred. Product failure data was captured in file
"FFDC.FTE.20100928170828514.8172766022149157013.log".
BFGSS0025E: An internal error has occurred. The exception is: cc=2 rc=2010 op=put - MQPUT to
SYSTEM.FTE.STATE.agent_name
BFGAG0061E: The agent ended abnormally
```

Les codes anomalie IBM MQ suivants peuvent être inclus dans ce message dans le journal des événements de l'agent:

- **rc=2010** Ce code anomalie est mappé à **MQRC_DATA_LENGTH_ERROR** et indique que la valeur de la taille maximale de message du canal client a été dépassée. Pour résoudre ce problème, assurez-vous que la taille de message maximale du canal client du gestionnaire de files d'attente de l'agent est supérieure ou égale au résultat du calcul suivant:

$3 * (\text{maxInputOutputMessageLength}) + 1048576$

- **rc=2030** Ce code anomalie est mappé à **MQRC_MSG_TOO_BIG_FOR_Q** et indique la valeur de la taille de message maximale de SYSTEM.FTE.STATE.*nom_agent* a été dépassée. Pour résoudre ce problème, assurez-vous que la taille de message maximale de SYSTEM.FTE.STATE. La file d'attente *nom_agent* est supérieure ou égale au résultat du calcul suivant:

$3 * (\text{maxInputOutputMessageLength}) + 1048576$

- rc=2031 Ce code anomalie est mappé à MQRC_MSG_TOO_BIG_FOR_Q_MGR et indique que la valeur de la taille de message maximale du gestionnaire de files d'attente de l'agent a été dépassée. Pour résoudre ce problème, assurez-vous que la taille de message maximale du gestionnaire de files d'attente de l'agent est supérieure ou égale au résultat du calcul suivant:

$$3 * (\text{maxInputOutputMessageLength}) + 1048576$$

Si vous transférez de nombreux messages de petite taille

Si la taille moyenne des messages que l'agent lit ou écrit dans une file d'attente est inférieure à 1310 octets et que l'agent lit ou écrit plus de 10000 messages, vous devez augmenter le nombre maximal d'attributs de messages non validés sur le gestionnaire de files d'attente ou réduire la quantité de données dans un intervalle de point de contrôle.

Lorsque l'agent lit ou écrit des messages dans une file d'attente, les **GET** ou **PUT** correspondants sont regroupés dans des transactions. Le nombre de **GET**s ou de **PUT**s dans une transaction est déterminé par le nombre requis pour traiter toutes les données dans un intervalle de point de contrôle. La quantité approximative des données dans un intervalle de point de contrôle est déterminée à partir des propriétés de l'agent à l'aide du calcul suivant:

$$\text{Checkpoint interval data size (in bytes)} = \text{agentCheckpointInterval} * \text{agentFrameSize} * \text{agentWindowSize} * \text{agentChunkSize}.$$

La taille des données de point de contrôle par défaut est $1 * 5 * 10 * 262144$ octets = 13107200 octets (12.5MB). Le nombre maximal de messages non validés dans une transaction pris en charge par un gestionnaire de files d'attente est contrôlé par l'attribut de gestionnaire de files d'attente **MaxUncommittedMsgs**. La valeur par défaut de cet attribut est 10000 messages. Si la taille moyenne des messages est inférieure à environ 1310 octets, le nombre maximal par défaut de messages non validés est dépassé s'il y a plus de 10000 messages à écrire.

Si vous dépassez la limite **MaxUncommittedMsgs**, l'agent s'arrête avec l'erreur suivante dans le journal des événements de l'agent:

```
BFGSS0024E: The agent has received a reason code of '2024' from the message queue interface (MQI).  
The agent cannot continue processing and will now end.  
BFGAG0139I: The agent has suspended its current transfers and is now stopping.
```

Le code anomalie 2024 est mappé à: MQRC_SYNCPOINT_LIMIT_ATTAINED.

Pour résoudre ce problème, effectuez l'une des actions suivantes:

- Augmentez la valeur de l'attribut de gestionnaire de files d'attente **MaxUncommittedMsgs** du gestionnaire de files d'attente auquel se connecte l'agent lors de la lecture ou de l'écriture dans une file d'attente. Voir [MaxUncommittedMsgs \(MQLONG\)](#).
- Réduisez la quantité de données dans un intervalle de point de contrôle. Pour ce faire, diminuez la valeur d'une ou de plusieurs des propriétés d'agent suivantes:
 - agentCheckpointInterval
 - agentFrameSize
 - agentWindowSize
 - agentChunkSize

Pour plus d'informations sur ces propriétés d'agent, voir [Propriétés d'agent avancées](#).

Si vous écrivez des messages dans une file d'attente de manière persistante

Si vous effectuez un transfert vers une file d'attente et que vous écrivez les messages dans la file d'attente de manière persistante, vous devrez peut-être augmenter la taille de l'espace fichier journal du gestionnaire de files d'attente pour pouvoir consigner toutes les données dans un intervalle de point de contrôle.

Si vous dépassez l'espace fichier journal du gestionnaire de files d'attente, l'agent s'arrête avec l'erreur suivante dans le journal des événements de l'agent:

```
BFGSS0024E: The agent has received a reason code of '2102' from the message queue interface (MQI).
The agent cannot continue processing and will now end.
BFGAG0062E: The agent has received MQI reason code '2102'. The agent cannot continue processing and
will now end.
BFGAG0061E: The agent ended abnormally
```

Le code anomalie '2102' est mappé à: MQRC_RESOURCE_PROBLEM.

Pour résoudre ce problème, augmentez la taille de l'espace fichier journal du gestionnaire de files d'attente de l'agent cible.

Conseils relatifs à l'utilisation du contrôle de compte utilisateur et du magasin virtuel avec MFT

Le contrôle de compte utilisateur (UAC) est présent dans Windows Server 2008 R2 et dans d'autres systèmes d'exploitation similaires. Il s'agit d'une infrastructure de sécurité et l'une de ses fonctions consiste à transférer les données utilisateur stockées dans le répertoire Program Files central vers un emplacement utilisateur, appelé magasin virtuel.

Si seuls les outils Managed File Transfer sont utilisés pour gérer les structures de données, Managed File Transfer n'est pas affecté par le contrôle de compte utilisateur et le magasin virtuel. Toutefois, si la structure de répertoire est modifiée ou reconstruite à l'aide d'outils de système d'exploitation standard par un administrateur non IBM MQ, il est possible que la nouvelle structure soit redirigée vers un magasin virtuel. Cela peut entraîner une ou plusieurs des situations suivantes:

- Les utilisateurs, y compris l'administrateur IBM MQ, ne peuvent plus voir les fichiers à l'emplacement prévu.
- Il se peut qu'un agent ne démarre pas et qu'il signale le message BFGCL0315, mais qu'il n'indique aucun code anomalie de prise en charge.
- Les fichiers journaux sont introuvables à l'emplacement indiqué par l'agent.
- Lorsqu'un agent est démarré avec le paramètre **-F**, il se peut qu'il ne démarre pas, ce qui génère le message suivant:

```
The current directory is invalid
```

Pour corriger toutes ces situations:

- En tant qu'administrateur IBM MQ, utilisez les commandes **fteDeleteAgent** et **fteCreateAgent** pour régénérer la structure de l'agent.
- En tant qu'administrateur du système d'exploitation, supprimez les entrées IBM MQ dans le magasin virtuel des utilisateurs concernés. Par exemple, sous Windows, l'emplacement du magasin virtuel est le suivant: `%USERPROFILE%\AppData\Local\VirtualStore\`

Référence associée

«**fteDeleteAgent**», à la page 473

La commande **fteDeleteAgent** supprime un Managed File Transfer Agent et sa configuration. Si l'agent est un agent de pont de protocole, le fichier de données d'identification de l'utilisateur est laissé sur le système de fichiers.

«Agent **fteCreate**(créer un agent MFT)», à la page 399

La commande **fteCreateAgent** crée un Managed File Transfer Agent et sa configuration associée.

Conseils pour l'exécution d'un agent ou d'un consigneur MFT en tant que service Windows

Vous pouvez exécuter un agent Managed File Transfer , un consigneur de base de données autonome et un consigneur de fichiers autonome, en tant que services Windows . Si vous rencontrez un problème avec ces services Windows , vous pouvez utiliser les fichiers journaux de service et les informations de cette rubrique pour diagnostiquer le problème.

Pour plus d'informations sur la configuration de votre agent, consigneur autonome ou consigneur de fichiers autonome pour une exécution en tant que service Windows , voir «[Démarrage d'un agent MFT en tant que service Windows](#)», à la page 159 et «[fteModifyLogger \(exécutez un consigneur MFT en tant que service Windows \)](#)», à la page 504.

Remarque : Si l'agent redistribuable doit s'exécuter en tant que service Windows, vous devez définir la variable d'environnement **BFG_DATA** dans l'environnement système pour que le service fonctionne.

Emplacement des fichiers journaux

Lorsque vous utilisez la commande **fteCreateAgent**, **fteCreateCDAgent**, **fteCreateBridgeAgent**, **fteModifyAgent**, **fteCreateLogger** ou **fteModifyLogger** pour exécuter un agent ou un consigneur en tant que service Windows , vous pouvez choisir le niveau de journalisation à l'aide du paramètre **-sl** . Les valeurs possibles pour ce paramètre sont `error`, `info`, `warn` et `debug`. La valeur par défaut est `info`.

Le fichier journal du service Windows porte le nom de fichier `servicedate.log`, où `date` correspond à la date de démarrage du service. Le fichier d'un agent est écrit dans le répertoire `MQ_DATA_PATH\mqft\logs\coordination_qmgr_name\agents\agent_name`. Ce répertoire est le même que celui dans lequel les fichiers de trace Managed File Transfer Agent sont écrits. Le fichier du consigneur est écrit dans le répertoire `MQ_DATA_PATH\mqft\logs\coordination_qmgr_name\loggers\logger_name`.

Si vous rencontrez des problèmes lors du démarrage d'un agent ou d'un consigneur autonome en tant que service Windows , essayez de définir le niveau de journalisation sur `debug` à l'aide du paramètre **-sl** . Des informations supplémentaires sont consignées dans le fichier `servicedate.log` .

Remarque : Lorsque le niveau de journalisation est défini sur `debug`, le compte utilisateur et le mot de passe que vous utilisez pour exécuter le service Windows sont affichés dans le fichier journal en texte en clair.

Nombre de fichiers journaux

Lorsque vous utilisez la commande **fteCreateAgent**, **fteCreateCDAgent**, **fteCreateBridgeAgent**, **fteModifyAgent**, **fteCreateLogger** ou **fteModifyLogger** pour exécuter un agent ou un consigneur autonome en tant que service Windows , vous pouvez choisir le nombre de fichiers journaux à l'aide du paramètre **-sj** . Spécifiez le texte suivant dans le cadre de votre commande pour modifier le nombre de fichiers journaux: `-sj -Dcom.ibm.wmqfte.daemon.windows.windowsServiceLogFiles=number`, où `number` est le nombre de fichiers journaux que vous souhaitez exprimer sous forme d'entier positif. Si vous ne spécifiez pas le nombre de fichiers journaux, la valeur par défaut est cinq.

Droit "Ouvrir une session en tant que service"

Le compte Windows que vous utilisez pour exécuter le service doit disposer du droit **Log on as a service** . Si vous tentez de démarrer le service, soit à l'aide de la commande **fteStartAgent**, **fteStartLogger** , soit à l'aide de la commande Windows **Sc.exe** , et que vous utilisez un compte

utilisateur qui ne dispose pas de ce droit, une fenêtre **Services** s'ouvre. Si le service que vous souhaitez démarrer doit exécuter un agent, cette fenêtre contient le message suivant:

```
Unable to start Windows service mqmftAgentAGENT@QMGR.  
System error 1069: The service did not start due to a logon failure.
```

Dans ce message, *AGENT* est le nom de votre agent et *QMGR* est le nom de votre gestionnaire de files d'attente d'agent. Si vous tentez d'exécuter un consignateur autonome en tant que service, un message similaire est généré, qui fait référence au consignateur plutôt qu'à un agent.

Pour éviter cette erreur, attribuez le droit **Log on as a service** au compte Windows que vous utilisez pour exécuter le service. Par exemple, sous Windows 7, procédez comme suit:

1. Dans le menu **Démarrer**, cliquez sur **Outils d'administration > Stratégie de sécurité locale**.
2. Dans le panneau **Paramètres de sécurité**, développez **Stratégies locales**, puis cliquez sur **Affectations de droits utilisateur**.
3. Dans le panneau **Paramètres de stratégie et de sécurité**, cliquez deux fois sur **Ouvrir une session en tant que service**.
4. Cliquez sur **Ajouter un utilisateur ou un groupe**, puis ajoutez l'utilisateur dont vous souhaitez exécuter le service à la liste des utilisateurs disposant du droit **Log on as a service**. Vous avez indiqué ce nom d'utilisateur lorsque vous avez exécuté la commande **fteCreateAgent**, **fteCreateCDAgent**, **fteCreateBridgeAgent**, **fteModifyAgent**, **fteCreateLogger** ou **fteModifyLogger**.

Remarque : L'erreur System error 1069: The service did not start due to a logon failure. peut également être due à un mot de passe incorrect.

Masquage du mot de passe de votre compte Windows

Lorsque vous configurez votre agent ou consignateur autonome pour qu'il s'exécute en tant que service Windows, vous spécifiez un nom d'utilisateur et un mot de passe à utiliser. Dans l'exemple suivant, l'agent *AGENT1* est créé, qui possède un gestionnaire de files d'attente d'agent *QMGR1* et qui est configuré pour s'exécuter en tant que service Windows :

```
fteCreateAgent -agentName AGENT1 -agentQMGr QMGR1 -s -su fteuser -sp ftepassword
```

Dans cet exemple, le service Windows s'exécute avec le nom d'utilisateur *fteuser*, auquel est associé un mot de passe *ftepassword*. Lorsque vous exécutez la commande **fteCreateAgent** ou l'une des autres commandes qui accepte le paramètre **-s**, vous spécifiez le mot de passe du compte Windows en texte en clair. Si vous préférez ne pas afficher votre mot de passe, procédez comme suit:

1. Exécutez la commande (**fteCreateAgent**, **fteCreateCDAgent**, **fteCreateBridgeAgent**, **fteModifyAgent**, **fteCreateLogger** ou **fteModifyLogger**) sans spécifier le paramètre **-sp**.
Exemple :

```
fteCreateAgent -agentName AGENT1 -agentQMGr QMGR1 -s -su fteuser
```

Remarque : La commande génère un message qui vous avertit que vous devez définir le mot de passe à l'aide de l'outil Windows Services avant que le service ne démarre correctement.

2. Ouvrez la fenêtre Windows **Services**.
3. Dans la liste des services, cliquez avec le bouton droit de la souris sur l'agent ou le service de consignateur autonome et sélectionnez **Propriétés**. Le nom d'affichage du service d'agent est Managed File Transfer Agent *AGENT @ QMGR*, où *AGENT* est le nom de l'agent et *QMGR* est le nom du gestionnaire de files d'attente de l'agent. Le nom d'affichage du service de consignateur est Managed File Transfer Logger for property set *coordination_qmgr_name*, où *coordination_qmgr_name* est le gestionnaire de files d'attente de coordination que vous avez spécifié pour le consignateur autonome à utiliser comme ensemble de propriétés. Pour plus d'informations

sur l'ensemble de propriétés, voir «[fteStartLogger](#)», à la page 542 et «[fteModifyLogger](#) (exécutez un consigneur MFT en tant que service Windows)», à la page 504.

4. Dans la fenêtre **Propriétés** , sélectionnez l'onglet **Connexion** .
5. Entrez le mot de passe du compte utilisateur qui exécute le service dans les zones **Mot de passe** et **Confirmer le mot de passe** . Les caractères du mot de passe sont masqués lorsque vous les entrez.
6. Cliquez sur **OK**.

Problèmes connus

Problème lors de l'utilisation de la variable d'environnement système **JAVA_HOME** (s'applique à **Managed File Transfer dans IBM WebSphere MQ 7.5.0 Fix Pack 1** ou **version antérieure uniquement**).

La variable d'environnement système **JAVA_HOME** ne doit pas être définie ; sinon, il est peu probable que le service Windows de l'agent ou du consigneur démarre. Le service Windows de l'agent ou du consigneur doit être exécuté avec l'environnement d'exécution IBM MQ Java .

Conseils pour la mise à jour des options JVM de l'agent ou du consigneur

Si vous utilisez le paramètre **-sj** de la commande **fteModifyAgent** ou **fteModifyLogger** pour modifier une définition de service Windows existante pour un agent ou un consigneur en mettant à jour, en ajoutant ou en supprimant des propriétés système Java , le service Windows existant est d'abord supprimé avant qu'un nouveau service ne soit créé à sa place, et le fichier de propriétés de l'agent ou du consigneur est mis à jour avec les propriétés du nouveau service Windows . La nouvelle définition de service Windows doit être cohérente avec les propriétés de service Windows mises à jour qui sont définies dans le fichier de propriétés de l'agent ou du consigneur.

Depuis la IBM MQ 9.0.0 Fix Pack 4, des vérifications supplémentaires sont ajoutées sous l'APAR IT22423 de sorte que toutes les mises à jour apportées aux options JVM pour un agent ou un consigneur avec le paramètre **-sj** de la commande **fteModifyAgent** ou **fteModifyLogger** soient vérifiées pour s'assurer que les options ont été correctement spécifiées. Si les propriétés sont jugées non valides ou qu'elles n'ont pas pu être validées, la commande **fteModifyAgent** ou **fteModifyLogger** échoue et un message d'erreur s'affiche.

Si les propriétés JVM sont valides et que la suppression du service Windows existant a abouti, mais qu'un échec se produit lorsque la commande **fteModifyAgent** ou **fteModifyLogger** crée le nouveau service Windows , la commande tente de supprimer les propriétés qui définissent le service Windows de remplacement du fichier de propriétés de l'agent ou du consigneur. Dans ce cas, des messages d'erreur sont renvoyés pour expliquer que l'agent ou le consigneur n'a pas pu être modifié, que l'ancien service Windows a été supprimé mais qu'un nouveau service Windows n'a pas pu être créé et que l'agent ou le consigneur ne s'exécutera donc pas en tant que service Windows . Vous devez ensuite vérifier manuellement que l'état de la définition de service Windows est cohérent avec les propriétés de service Windows qui sont définies dans le fichier de propriétés de l'agent ou du consigneur, et prendre les mesures appropriées pour corriger les éventuelles incohérences.

Information associée

[Agent fteModify: exécutez un agent MFT en tant que service Windows](#)

[fteModifyLogger: exécutez un consigneur MFT en tant que service Windows](#)

Conseils pour la configuration d'un moniteur de ressources MFT afin d'éviter la surcharge d'un agent

Vous pouvez configurer les valeurs de propriété et de paramètre d'un moniteur de ressources Managed File Transfer pour réduire la charge sur un agent. La réduction de la charge sur l'agent améliore les performances de cet agent. Il existe plusieurs paramètres que vous pouvez utiliser, et vous devrez peut-être utiliser des essais et des erreurs pour trouver les meilleurs paramètres pour votre configuration système.

Présentation de la surveillance des ressources

Lorsqu'un moniteur de ressources interroge un répertoire ou une file d'attente, l'agent effectue les étapes suivantes:

- Recherche tous les fichiers correspondant à un modèle de déclencheur (par exemple, tous les fichiers *.txt du répertoire). Ou recherche tous les groupes complets de messages dans la file d'attente.
- Détermine les fichiers qui sont nouveaux ou modifiés, ou détermine les groupes qui sont nouveaux dans la file d'attente.
- Lance des transferts pour les fichiers ou les groupes qui correspondent aux critères des deux étapes précédentes.
- S'ajoute à la liste des fichiers et des groupes déjà transférés afin qu'ils ne soient plus transférés tant qu'ils n'ont pas été modifiés.

Pour un moniteur de répertoire, plus il y a de fichiers dans le répertoire source et plus le modèle de déclenchement est large, plus la liste des fichiers que l'agent doit analyser et comparer à la liste des fichiers déjà transférés est grande.

Pour un moniteur de file d'attente, plus le nombre de groupes dans la file d'attente est élevé, plus la liste des groupes que l'agent doit comparer à la liste des groupes déjà transférés est longue.

Prenez en compte les paramètres de clé suivants:

- Utilisez la propriété d'agent **monitorMaxResourcesInPoll** pour définir le nombre maximal de fichiers ou de groupes que l'agent inclut dans chaque interrogation. L'utilisation de ce paramètre limite le nombre de transferts dans un intervalle d'interrogation. Cela signifie également que l'agent a moins d'analyse syntaxique à effectuer avant d'initier un transfert pour ce nombre de fichiers ou de groupes. Lors de la prochaine interrogation du moniteur de répertoire ou du moniteur de file d'attente, l'agent inclut l'ensemble de fichiers ou de groupes suivant. La propriété d'agent **monitorMaxResourcesInPoll** est disponible dans IBM WebSphere MQ File Transfer Edition 7.0.4 et les versions ultérieures. Pour les versions antérieures de IBM WebSphere MQ File Transfer Edition, elle est disponible en tant que correctif temporaire pour l'APAR IC78011.
- Lors de la création d'un moniteur de répertoire, assurez-vous que la définition de transfert que vous configurez possède une disposition source de `delete`. La définition de cette disposition signifie que lorsque le transfert de fichiers est terminé, il est supprimé du répertoire contrôlé et l'agent ne le conserve plus dans sa liste interne.
- Lors de la création d'un moniteur de répertoire, utilisez le paramètre **-rl** dans la commande **fteCreateMonitor** pour limiter le nombre de niveaux du répertoire que l'agent doit parcourir. L'utilisation de ce paramètre signifie que les répertoires de niveau inférieur ne sont pas analysés inutilement.

Autres considérations lors de la création d'un moniteur de ressources

Le processus d'interrogation du moniteur de ressources consomme des ressources d'agent. L'augmentation de l'intervalle d'interrogation d'un moniteur réduit la charge placée sur l'agent. Toutefois, la définition de l'intervalle d'interrogation doit être équilibrée par rapport à la génération d'un trop grand nombre de transferts par intervalle d'interrogation. Tenez compte des points suivants lorsque vous définissez l'intervalle d'interrogation d'un moniteur de ressources:

- Vitesse à laquelle vous avez besoin d'un transfert à lancer après qu'un fichier a été placé dans un répertoire ou un groupe dans une file d'attente.
- Fréquence à laquelle les fichiers sont placés dans un répertoire ou des groupes dans une file d'attente.
- Vitesse de transfert maximale de l'agent. L'agent doit pouvoir gérer tous les transferts générés par un moniteur.

L'intervalle d'interrogation est spécifié lorsque le moniteur de ressources est créé à l'aide de la commande **fteCreateMonitor** en spécifiant les paramètres **-pi** (intervalle d'interrogation) et **-pu**

(unités d'intervalle d'interrogation). Vous devrez peut-être expérimenter pour déterminer les meilleurs paramètres de votre configuration.

Une option permettant d'améliorer la stabilité des agents à haute charge qui exécutent des moniteurs de ressources consiste à réduire la valeur de la propriété d'agent `maxSourceTransfers`. Avec cette option, l'agent répartit le temps de traitement entre le moniteur de ressources et le transfert de fichiers. Plus la valeur de la propriété d'agent `maxSourceTransfers` est élevée, plus le temps de traitement est consommé par le transfert de fichiers et moins il est disponible pour le moniteur de ressources. Si vous réduisez la valeur de la propriété d'agent `maxSourceTransfers`, l'agent effectue moins de transferts en parallèle, mais il doit disposer de suffisamment de temps de traitement pour interroger ses moniteurs de ressources. Si vous réduisez la valeur de cette propriété d'agent, vous devez envisager d'augmenter la valeur de la propriété d'agent `maxQueuedTransfers` car le nombre de transferts en file d'attente peut augmenter.

Si, après avoir optimisé votre moniteur, vous constatez que certains transferts entrent dans la récupération, envisagez d'augmenter la valeur du délai d'attente de l'agent. La charge élevée placée sur l'agent peut signifier que les transferts expirent lors de la négociation du début du transfert avec l'agent de destination. Ce délai entraîne la reprise du transfert et retarde la fin du transfert. La propriété d'agent `maxTransferNegotiationTime` indique le temps pendant lequel l'agent source attend une réponse de l'agent de destination. Si ce délai est dépassé, le transfert est récupéré. La valeur par défaut de cette propriété est 30000 millisecondes (30 secondes). L'augmentation de la valeur de la propriété, par exemple à 300000 millisecondes (5 minutes), peut permettre la poursuite des transferts sans dépassement du délai d'attente et éviter la reprise.

Concepts associés

[«Utilisation de fichiers de définition de transfert», à la page 164](#)

Vous pouvez spécifier un fichier de définition de transfert qui peut être utilisé pour créer un transfert de fichier. Le fichier de définition de transfert est un fichier XML qui définit tout ou partie des informations requises pour créer le transfert.

Tâches associées

[«Surveillance des ressources MFT», à la page 174](#)

Vous pouvez surveiller les ressources Managed File Transfer ; par exemple, une file d'attente ou un répertoire. Lorsqu'une condition est satisfaite sur cette ressource, le moniteur de ressources démarre une tâche, telle qu'un transfert de fichier. Vous pouvez créer un moniteur de ressources à l'aide de la commande `fteCreateMonitor` ou de la vue **Moniteurs** dans le plug-in Managed File Transfer pour IBM MQ Explorer.

Référence associée

[«fteCreateMonitor: créer un moniteur de ressources MFT», à la page 424](#)

La commande `fteCreateMonitor` crée et démarre un nouveau moniteur de ressources à partir de la ligne de commande. Vous pouvez surveiller une ressource (par exemple, le contenu d'un répertoire) à l'aide de Managed File Transfer de sorte que lorsqu'une condition de déclenchement est remplie, une tâche spécifiée, telle qu'un transfert de fichier, est démarrée.

Que faire si la substitution de variables entraîne l'accès de plusieurs fichiers à un même nom de fichier

Pour Managed File Transfer, si vous surveillez un répertoire et transférez plusieurs fichiers d'une source vers un emplacement de destination et que vous utilisez la substitution de variable `${FileName}`, vous devez tester les résultats de la substitution de variable. Les résultats doivent être testés car l'utilisation de la substitution de variables peut entraîner l'appel de combinaisons inattendues de commandes de transfert de fichiers.

Pour déterminer si le problème se produit, recherchez les cas où plusieurs fichiers semblent être transférés, mais où un seul fichier arrive à la destination. Des erreurs peuvent apparaître dans le journal de transfert de fichiers indiquant que plusieurs fichiers tentent d'effectuer un transfert vers le même nom de fichier cible et que les transferts échouent vers le même nom de fichier.

Pourquoi ce problème se produit

Lorsque plusieurs fichiers sont traités par un moniteur de répertoire MFT, le fichier xml de tâche s'exécute pour chaque fichier détecté par le moniteur dans le répertoire surveillé. Si `${FileName}` est spécifié uniquement dans la destination du fichier de tâche xml et non dans la source, le transfert est appelé pour chaque fichier plusieurs fois, une fois pour chaque combinaison de noms de fichier.

Exemple :

```
<source disposition="delete" recursive="false">
  <file>e:\temp</file>
</source>
<destination exist="overwrite" type="file">
  <file>s:\outdir\${FileName}</file>
</destination>
```

Eviter ce problème

Si vous utilisez la substitution de variable `${FileName}` dans la source ou la destination et que vous vous attendez à ce qu'une variante du même nom de fichier arrive à la destination, veillez à spécifier `${FileName}` dans la source et la destination de votre définition XML de tâche.

L'exemple suivant prend un fichier de `e:\temp\<filename>` et le transfère à `s:\outdir\<filename>.out`:

```
<source disposition="delete" recursive="false">
  <file>e:\temp\${FileName}</file>
</source>
<destination exist="overwrite" type="file">
  <file>s:\outdir\${FileName}.out</file>
</destination>
```

Concepts associés

[«Personnalisation des tâches du moniteur de ressources MFT avec la substitution de variables»](#), à la page 185

Lorsque les conditions de déclenchement d'un moniteur de ressources actif sont satisfaites, la tâche définie est appelée. En plus d'appeler la tâche de transfert ou de commande avec le même agent de destination ou le même nom de fichier de destination à chaque fois, vous pouvez également modifier la définition de la tâche lors de l'exécution. Pour ce faire, vous devez insérer des noms de variable dans le fichier XML de définition de tâche. Lorsque le moniteur détermine que les conditions de déclenchement sont satisfaites et que la définition de tâche contient des noms de variable, il remplace les noms de variable par les valeurs de variable, puis appelle la tâche.

[«Exemples: substitution de variable pour les définitions de moniteur de ressources»](#), à la page 188

Exemples de substitution de variables pour les définitions de moniteur de ressources à l'aide de XML et de IBM MQ Explorer.

Que faire si vous recevez une erreur lors de la mise à jour de votre schéma de base de données MFT sur une base de données Oracle

Vous pouvez recevoir le message d'erreur suivant lors de la mise à jour de votre schéma de base de données vers le niveau le plus récent à l'aide du fichier `ftelog_tables_oracle_702_703.sql` :
ERROR at line 1: ORA-02289: sequence does not exist. Cette erreur se produit car les séquences et les déclencheurs utilisés par les tables ne se trouvent pas dans le même schéma que les tables.

Pourquoi et quand exécuter cette tâche

Pour résoudre ce problème, vous devez éditer le contenu du `ftelog_tables_oracle_702_703.sql` avant de l'exécuter.

Procédure

1. Identifiez le schéma dans lequel se trouvent les séquences et les déclencheurs utilisés par les tables du consignateur de base de données Managed File Transfer .
 - Sous Db2, vous pouvez utiliser le Control Center pour afficher les tables et le schéma.
 - Sous Oracle, vous pouvez utiliser Enterprise Manager pour afficher les tables et le schéma.
2. Ouvrez le fichier `ftelog_tables_oracle_702_703.sql` dans n'importe quel éditeur de texte.
3. Dans chaque occurrence du texte, `SELECT FTELOG.sequence_name.nextval` remplacez le texte `FTELOG` par le nom du schéma dans lequel se trouvent vos séquences existantes.
4. Avant chaque occurrence du texte `CREATE OR REPLACE TRIGGER FTELOG.trigger_name`, insérez le texte `DROP TRIGGER schema_name.trigger_name`, où `schema_name` est le nom du schéma dans lequel se trouvent vos déclencheurs existants.
5. Utilisez le fichier `ftelog_tables_oracle_702_703.sql` modifié pour mettre à jour les tables de base de données.

Traitement et rejet des erreurs du consignateur MFT

Le consignateur Managed File Transfer identifie deux types d'erreur: les erreurs par message et les erreurs générales.

Les erreurs par message sont probablement dues à un problème avec un ou plusieurs messages individuels. Voici quelques exemples de situations identifiées comme des erreurs par message:

- Le code de résultat, qui est un élément de données obligatoire, est manquant dans un message
- Un transfert indique un nom de travail de 3000 caractères et trop grand pour la colonne de base de données associée
- Un message de progression est reçu pour un transfert, mais il n'y a aucun enregistrement du début du transfert (peut-être en raison d'un message de début de transfert mal acheminé ou retardé)
- Un message est reçu, qui n'est pas un message de journal Managed File Transfer

Les erreurs générales sont toutes les erreurs qui ne sont pas des erreurs par message. Ces erreurs sont probablement dues à des problèmes de configuration ou à des erreurs de programme.

Lorsqu'une erreur par message est détectée, le consignateur rejette le message en plaçant le message dans la file d'attente de rejet. Rien n'est consigné dans le journal de sortie. Par conséquent, inspectez régulièrement ou surveillez en permanence la file d'attente de rejet pour détecter les messages rejetés.

Si un trop grand nombre de messages sont rejetés consécutivement, sans qu'aucun message n'ait été correctement écrit dans la base de données, cette erreur est traitée comme une erreur générale. Par exemple, imaginez un site qui utilise toujours des codes à 10 caractères comme noms de travail, mais qui a reconfiguré par inadvertance la colonne de nom de travail pour qu'elle comporte deux caractères. Bien que les données trop larges soient généralement une erreur par message, dans ce cas, le problème de configuration est général et est détecté comme une erreur générale. Vous pouvez optimiser le nombre d'erreurs consécutives par message nécessaires pour provoquer une erreur générale à l'aide de la propriété **`wmqfte.max.consecutive.reject`**.

Si une erreur générale est détectée, le consignateur annule tous les messages non encore validés dans le gestionnaire de files d'attente, puis effectue des tentatives périodiques. Un message identifiant le problème est consigné dans le journal de sortie et dans la console si le consignateur a été démarré en mode avant-plan avec le paramètre **`-F`**.

L'emplacement des journaux de sortie du consignateur varie selon qu'il s'agit d'un consignateur de base de données JEE ou autonome. Pour un consignateur de base de données autonome, il se trouve dans le répertoire `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/loggers/logger_name`. Pour un consignateur de base de données JEE, il se trouve dans le journal de sortie standard du serveur d'applications.

File d'attente de rejet

Les messages qui génèrent des erreurs par message sont déplacés dans la file d'attente de rejet. Sur chaque message rejeté, une propriété de message est définie pour indiquer la raison pour laquelle le message a été rejeté. Le nom complet de la propriété est **usr.WMQFTE_ReasonForRejection**, bien qu'il soit utilisé. est omis dans certains contextes (notamment JMS et IBM MQ Explorer).

Si vous utilisez IBM MQ Explorer, vous pouvez afficher le contenu de la file d'attente de rejet en cliquant avec le bouton droit de la souris sur la file d'attente et en cliquant sur **Parcourir les messages**. Pour savoir pourquoi un message a été rejeté, cliquez deux fois sur le message pour ouvrir sa boîte de dialogue de propriétés, puis sélectionnez la page **Propriétés nommées**. Vous verrez une propriété appelée **WMQFTE_ReasonForRejection**. Vous pouvez également écrire ou configurer un outil de surveillance pour obtenir ces informations automatiquement.

Il peut arriver que vous souhaitiez retraiter des messages de la file d'attente de rejet. Dans l'exemple décrit précédemment dans cette rubrique, avec une colonne de nom de travail à deux caractères dans la base de données, les messages peuvent être traités avec succès après l'augmentation de la largeur de la colonne de base de données. Comme autre exemple, lorsqu'un message de fin de transfert est rejeté car le début de transfert associé était manquant, le message de début de transfert peut être reçu ultérieurement. Le retraitement du transfert sera alors effectué avec succès.

Pour retraiter les messages, déplacez-les de la file de rejet vers la file d'entrée. Dans une installation normale, où le consignateur a créé son propre abonnement géré, la file d'attente d'entrée est définie par le gestionnaire de files d'attente et a un nom tel que SYSTEM.MANAGED.DURABLE.49998CFF20006204. Vous pouvez identifier la file d'entrée en consultant le **Nom de la destination** dans les propriétés de l'abonnement SYSTEM.FTE.DATABASELogger.AUTO ou à l'aide de la commande MQSC suivante:

```
DISPLAY SUB(SYSTEM.FTE.DATABASELogger.AUTO) DEST
```

Une façon de déplacer des messages entre des files d'attente consiste à utiliser [MA01 SupportPac](#), par exemple:

```
q -IFTE.REJECT -oSYSTEM.MANAGED.DURABLE.49998CFF20006204
```

La file d'attente de rejet peut contenir des messages rejetés pour diverses raisons, dont certaines seulement ont été résolues. Dans ce cas, vous pouvez toujours retraiter tous les messages ; les messages qui peuvent maintenant être acceptés sont consommés et les messages qui ne peuvent pas être déplacés à nouveau dans la file d'attente de rejet.

Les messages de journal Malformed dans le journal de transfert ne sont pas consignés par le consignateur. Ces messages n'étant pas considérés comme significatifs, ils sont envoyés à la file d'attente de rejet. Pour plus d'informations sur les messages du journal de transfert, voir [«Formats des messages du journal de transfert de fichiers»](#), à la page 771.

Que faire si le consignateur MFT est démarré, mais qu'aucune information de transfert n'est consignée dans la base de données

Les tables de base de données utilisées par le consignateur Managed File Transfer requièrent que la taille de page de la base de données soit supérieure ou égale à 8 Ko. Si la taille de page de la base de données n'est pas assez grande, les tables ne sont pas créées correctement et l'erreur SQLSTATE=42704s'affiche.

Si vous utilisez le consignateur de base de données Java Platform, Enterprise Edition, vous pouvez voir le message suivant dans le journal de sortie du système WebSphere Application Server ; si vous utilisez le consignateur de base de données autonome, vous pouvez voir l'erreur suivante dans le fichier output0.log :

```
DB2 SQL ERROR: SQLCODE=-204, SQLSTATE=42704  
SQLERRMC=FTELOG.TRANSFER_EVENT, DRIVER=3.40.152
```

La valeur SQLSTATE de 42704 indique qu'il s'agit d'une table que le consignateur s'attendait à voir exister, en l'occurrence FTELOG.TRANSFER_EVENT, n'existe pas.

Pour résoudre ce problème, procédez comme suit:

1. Vérifiez que la table existe et qu'elle est complète. Pour plus d'informations sur les tables utilisées par le consignateur et leurs colonnes, voir «Tables du consignateur de base de données MFT», à la page 688.
2. Si la table n'existe pas ou est incomplète, vérifiez la taille de page de la base de données.
3. Si la taille de la base de données est inférieure à 8 Ko, augmentez la taille de page de votre base de données.
 - Si votre base de données se trouve sur un système de test ou ne contient pas de données, vous pouvez supprimer les tables et recréer la base de données avec une taille de page supérieure à 8 Ko.
 - Pour plus d'informations sur l'augmentation de la taille de page, voir «Augmentation de la taille de page de la base de données de journaux sous Db2 sous Windows, UNIX ou Linux», à la page 20  ou «Migration des tables de base de données sous Db2 sous z/OS vers IBM MQ 8.0», à la page 22.

Que faire si MFT ne lit pas les propriétés du magasin de clés à partir du fichier de configuration du magasin de clés dans AMS

L'emplacement du fichier de configuration du magasin de clés, s'il n'est pas présent dans l'emplacement par défaut, doit être spécifié par la variable `MQS_KEYSTORE_CONF` pour que Java AMS s'exécute en mode client. Si l'emplacement n'est pas spécifié, les journaux Managed File Transfer Agent affichent le message d'erreur: "Failed to read keystore properties from the keystore configuration file."

L'emplacement par défaut du fichier de configuration du magasin de clés est `home_directory/.mqsf/keystore.conf`. Si l'emplacement du fichier de configuration du magasin de clés n'est pas l'emplacement par défaut, procédez comme suit:

1. Démarrez l'agent FTE en mode client.
2. Appliquez la sécurité AMS à SYSTEM.FTE.DATA. < nom de l'agent > file d'attente. Si le fichier de configuration du magasin de clés ne se trouve pas à cet emplacement, tous les transferts échoueront sans accusé de réception.
3. Définissez la variable système `BFG_JVM_PROPERTIES` sur `BFG_JVM_PROPERTIES=-DMQS_KEYSTORE_CONF=chemin d'accès au fichier keystore_config` pour la commande `fteStartAgent`.
4. Définissez la variable système `MQS_KEYSTORE_CONF` sur `MQS_KEYSTORE_CONF=chemin d'accès au fichier keystore_config` pour la commande `fteStartAgent`. Ce paramètre doit être défini pour garantir l'exécution de tous les agents, quel que soit le mode dans lequel ils s'exécutent.

Remarque : Si Java AMS s'exécute en mode liaisons, l'erreur AMQ9062 s'affiche dans le journal des erreurs du gestionnaire de files d'attente si le fichier de configuration du magasin de clés ne se trouve pas à l'emplacement par défaut.

Que faire si votre agent MFT ABENDS avec une erreur `java.lang.OutOfMemoryError` en raison d'un épuisement de la mémoire native

Lors du traitement d'un certain nombre de demandes de transfert gérées, telles que les transferts de fichier à fichier, de message à fichier ou de fichier à message, l'agent se termine de manière anormale (ABENDS) en signalant un `java.lang.OutOfMemoryError`, et au moment où votre mémoire RAM totale n'était pas entièrement utilisée. Cette exception a été provoquée par l'épuisement de la mémoire native.

Diagnostic du problème

Lorsque ce problème se produit, l'agent concerné se termine de la sorte et génère deux fichiers qui fournissent des détails sur la cause première:

- Fichier ABEND. Le nom de ce fichier est conforme à la convention de dénomination `ABEND.FTE.date_timestamp.identifrier.log`.

► **Multi** Sur Multiplatforms, le fichier est écrit dans le répertoire `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name/logs/ffdc`.

► **z/OS** Sous z/OS, le fichier est écrit dans l'emplacement `USS $BFG_CONFIG/mqft/logs/coordination_qmgr_name/agents/agent_name/logs/ffdc`

- Un fichier Javacore. Le nom de ce fichier est au format suivant: `javacore.datestamp.timestamp.pid.identifrier.txt`

► **Multi** Sur Multiplatforms, le fichier est écrit dans le répertoire `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name`.

► **z/OS** Sous z/OS, le fichier est écrit dans le répertoire `$BFG_CONFIG/mqft/logs/coordination_qmgr_name/agents/agent_name` de l'emplacement USS.

La paire ABEND et Javacore contient des informations similaires aux exemples ci-dessous:

Exemple: paire 1

Fichier de fin anormale

```
Filename:
C:\ProgramData\IBM\MQ\mqft\logs\COORDQM\agents\AGENT1\logs\ffdc\ABEND.FTE.20200109113518046.1764802189777
906538.log
Level: p900-005-180821
Time: 09/01/2020 11:35:18:046 GMT
Thread: 96 (TransferSender[414d51204d4424b525030372020202045fbd6532ebfaa02])
Class: com.ibm.wmqfte.thread.FTEThread
Instance: 55b455b4
Method: uncaughtException
Probe: ABEND_001
Cause: java.lang.OutOfMemoryError: native memory exhausted

java.lang.OutOfMemoryError: native memory exhausted
at com.ibm.mq.jmqi.local.internal.base.Native.MQPUT(Native Method)
at com.ibm.mq.jmqi.local.LocalMQ.MQPUT(LocalMQ.java)
at com.ibm.wmqfte.wmqiface.WMQQueueImpl.put(WMQQueueImpl.java)
at com.ibm.wmqfte.wmqiface.WMQQueueImpl.put(WMQQueueImpl.java)
at com.ibm.wmqfte.transfer.impl.TransferSenderRunnable.doTransfer(TransferSenderRunnable.java)
at com.ibm.wmqfte.transfer.impl.TransferSenderRunnable.run(TransferSenderRunnable.java)
at java.lang.Thread.run(Thread.java)
at com.ibm.wmqfte.thread.FTEThread.run(FTEThread.java)
```

Fichier Javacore

```
NULL -----
0SECTION TITLE subcomponent dump routine
NULL =====
1TISIGINFO Dump Event "systhrow" (00040000) Detail "java/lang/OutOfMemoryError" "native memory
exhausted" received
1TIDATETIME Date: 2020/01/09 at 11:35:18
1TIFILENAME Javacore filename:
C:\ProgramData\IBM\MQ\mqft\logs\COORDQM\agents\AGENT1\javacore.20200109.113518.14148.0002.txt
```

Exemple: paire deux

Fichier ABEND

```
Filename:
C:\ProgramData\IBM\MQ\mqft\logs\COORDQM\agents\AGENT1\logs\ffdc\ABEND.FTE.20200109143700286.3177895731698
464509.log
```

```
Level: p900-005-180821
Time: 09/01/2020 14:37:00:286 GMT
Thread: 918 (AgentStatusPublisher)
Class: com.ibm.wmqfte.thread.FTETHread
Instance: bc10bc1
Method: uncaughtException
Probe: ABEND_001
Cause: java.lang.OutOfMemoryError: Failed to create a thread: retVal -1073741830, errno 12
```

```
java.lang.OutOfMemoryError: Failed to create a thread: retVal -1073741830, errno 12
at java.lang.Thread.startImpl(Native Method)
at java.lang.Thread.start(Thread.java)
```

Fichier Javacore

```
NULL -----
0SECTION TITLE subcomponent dump routine
NULL =====
1TISIGINFO Dump Event "systhrow" (00040000) Detail "java/lang/OutOfMemoryError" "Failed to create a
thread: retVal -1073741830, errno 12" received
1TIDATETIME Date: 2020/01/09 at 14:37:00
1TIFILENAME Javacore filename: C
C:\ProgramData\IBM\MQ\mqft\logs\C00RDQM\agents\AGENT1\javacore.20200109.143700.2652.0003.txt
```

Pourquoi ce problème se produit

Ce problème se produit en raison de l'épuisement du segment de mémoire natif sur le système où l'agent est en cours d'exécution.

Pour plus d'informations sur les distinctions entre le segment de mémoire Java et le segment de mémoire natif, voir [«Comment les agents MFT utilisent le segment de mémoire Java et le segment de mémoire natif»](#), à la page 755 .

Eviter le problème

Vous pouvez effectuer un certain nombre d'actions pour réduire la probabilité qu'un agent MFT s'arrête en raison d'un `java.lang.OutOfMemoryError`, en raison de l'épuisement de la mémoire native:

1. Réduisez la taille du segment de mémoire Java pour la machine virtuelle Java exécutant l'agent MFT .

Plus la taille du segment de mémoire Java alloué est grande, moins le segment de mémoire natif dispose de mémoire. La réduction de la taille du segment de mémoire Java utilisé par un agent peut libérer plus de mémoire pour le segment de mémoire natif.

Par défaut, le segment de mémoire Java d'un agent est défini sur 512 Mo. Si vous avez modifié cette valeur pour la rendre plus importante, envisagez de la réduire et de la tester avec votre charge de travail de type production.

- Pour réduire ou modifier le segment de mémoire Java lors de l'exécution de l'agent en tant que processus normal:

Définissez la variable d'environnement `BFG_JVM_PROPERTIES` pour transmettre le répertoire d'options à la machine virtuelle Java. Par exemple, sous Windows, pour définir la taille de segment de mémoire maximale à 1024 Mo, exécutez la commande suivante avant d'utiliser la commande **`fteStartAgent`** :

```
set BFG_JVM_PROPERTIES="-Xmx1024M"
```

Pour plus d'informations sur la définition des propriétés système Java à l'aide de la variable d'environnement `BFG_JVM_PROPERTIES`, voir [Propriétés systèmeJava pour MFT](#).

- Pour réduire ou modifier le segment de mémoire Java lors de l'exécution de l'agent en tant que service Windows :

Pour transmettre des options à la machine virtuelle Java qui exécute l'agent en tant que service Windows , modifiez l'agent à l'aide du paramètre **`-sj`** spécifié dans la commande **`fteModifyAgent`** .

L'exemple suivant utilise la commande **fteModifyAgent** avec le paramètre **-sj** pour définir la taille maximale du segment de mémoire Java pour une machine virtuelle Java exécutant un agent configuré de service Windows :

```
fteModifyAgent.cmd -agentName AGENT1 -s -su user1 -sp passwd -sj -Xmx1024M
```

Vous pouvez vérifier que cette valeur a été correctement définie, en consultant le fichier `output0.log` de l'agent, une fois que l'agent a été redémarré. Dans la section *Démarrer l'affichage de l'environnement en cours*, une valeur de 1024 Mo est indiquée, comme suit:

```
The maximum amount of memory that the Java virtual machine will attempt to use is: '1024'MB
```

2. Restreindre l'utilisation de la mémoire native

Souvent, `java.lang.OutOfMemoryErrors` causé par l'épuisement du segment de mémoire natif est visible si un agent se connecte à son gestionnaire de files d'attente d'agent à l'aide du transport BINDINGS. Lorsque l'agent a été configuré pour utiliser le transport BINDINGS, il appelle des méthodes natives chaque fois qu'il doit communiquer avec le gestionnaire de files d'attente.

Cela signifie que l'utilisation de la mémoire native augmente à mesure que la charge de travail de l'agent augmente, en raison d'un plus grand nombre de connexions au gestionnaire de files d'attente et d'une meilleure communication des messages. Dans ce cas, la réduction de la charge de travail peut être utile. Pour ce faire, définissez les propriétés d'agent suivantes sur une valeur inférieure à la valeur par défaut 25:

- **maxSourceTransfers**
- **maxDestinationTransfers**

Cela réduit le nombre de transferts simultanés qui peuvent se produire et, par conséquent, diminue la charge de travail simultanée maximale pour l'agent.

3. Configurez l'agent pour qu'il utilise le transport CLIENT lors de la connexion à son gestionnaire de files d'attente d'agent. Pour ce faire, vous pouvez définir les propriétés d'agent suivantes:

- **agentQMgrHost**
- **agentQMgrPort**
- **agentQMgrChannel**

Vous trouverez des informations sur ces propriétés dans la rubrique [The MFT agent.properties file](#).

Cela garantit que toutes les communications entre l'agent et le gestionnaire de files d'attente s'effectuent sur TCP/IP, plutôt que sur du code natif, ce qui réduit la quantité de mémoire native utilisée par l'agent.

Important : Cette action diminue également les performances. En utilisant une connexion TCP/IP à l'hôte local, plutôt qu'un code natif, la configuration n'est pas aussi efficace lorsque l'agent requiert des interactions avec le gestionnaire de files d'attente.

Erreurs BFGSS0023E et comment les éviter

Si vous désinstallez un groupe de correctifs d'une installation afin de revenir à une version précédente du produit et qu'un agent associé à l'installation était impliqué dans des transferts gérés au moment de la désinstallation, cet agent ne peut pas démarrer et signale une erreur BFGSS0023E. Vous pouvez éviter cette erreur en effectuant un certain nombre d'étapes qui doivent empêcher l'affichage des messages BFGSS0023E lors du redémarrage des agents.

Pour chaque transfert géré en cours dans lequel un agent est actuellement impliqué, un message apparaît sur le système `SYSTEM.FTE.STATE.nom_agent` queue. Ce message stocke les informations de point de contrôle sur le transfert géré et est utilisé si le transfert géré fait l'objet d'une reprise. Une fois qu'un transfert géré est terminé, le message correspondant apparaît sur le système `SYSTEM.FTE.STATE`. La file d'attente `nom_agent` est supprimée.

Chaque message d'état contient des informations d'en-tête internes indiquant la version du composant Managed File Transfer utilisée par un agent lors de l'exécution du transfert géré. Les informations de version indiquent le niveau de groupe de correctifs spécifique. Par exemple, si un agent IBM MQ 8.0.0 Fix Pack 5 exécutait un transfert géré, le message d'état de ce transfert géré contient une référence à IBM MQ 8.0.0 Fix Pack 5.

Si un groupe de correctifs est désinstallé d'une installation et qu'un agent associé à cette installation est associé à des transferts en cours, l'agent ne démarre pas et signale l'erreur suivante:

BFGSS0023E: L'agent est configuré pour utiliser des files d'attente IBM MQ qui contiennent des données créées à l'aide d'une version ultérieure du produit. L'agent ne peut pas s'exécuter dans cette configuration et va s'arrêter.

Par exemple, si un agent IBM MQ 8.0.0 Fix Pack 5 a des transferts en cours d'exécution lorsqu'il est arrêté puis rétromigré vers le niveau IBM MQ 8.0.0 Fix Pack 4, lors du prochain démarrage de l'agent, il vérifie les messages sur son `SYSTEM.FTE.STATE.nom_agent` et détecte qu'ils ont été écrits lors de l'utilisation de IBM MQ 8.0.0 Fix Pack 5. Comme il utilise maintenant IBM MQ 8.0.0 Fix Pack 4, l'agent signale l'erreur BFGSS0023E décrite dans le paragraphe précédent et s'arrête.

En règle générale, si vous souhaitez supprimer un groupe de correctifs du composant Managed File Transfer, l'exécution des étapes suivantes doit empêcher l'affichage des messages BFGSS0023E lors du redémarrage des agents:

1. Assurez-vous que tous leurs agents ont terminé leurs transferts gérés.
2. Arrêtez les agents.
3. Supprimez le groupe de correctifs.
4. Redémarrez les agents.

Tâches associées

«Démarrage d'un agent MFT», à la page 158

Avant de pouvoir utiliser un agent Managed File Transfer pour un transfert de fichier, vous devez d'abord démarrer l'agent.

Référence associée

«Paramètres de file d'attente de l'agent MFT», à la page 637

Les scripts de commande MQSC générés par la commande **fteCreateAgent** créent les files d'attente d'agent avec les paramètres définis sur les valeurs suivantes. Si vous n'utilisez pas les scripts MQSC fournis pour créer les files d'attente, mais que vous les créez manuellement, veillez à définir les paramètres suivants sur les valeurs indiquées.

Information associée

[BFGSS0001 - BFGSS9999](#)

[Rétablissement d'une version précédente d'un gestionnaire de files d'attente sous UNIX](#)

[Rétablissement d'une version précédente d'un gestionnaire de files d'attente sous Windows](#)

Que faire si les transferts gérés échouent avec des erreurs BFGIO0341E

Si un transfert géré transfère un fichier dans un emplacement surveillé par un processus externe, il est possible que ce transfert géré échoue avec l'erreur: BFGIO0341E: Le renommage du fichier temporaire `nom_fichier_destination.part` en `nom_fichier_destination` a échoué car le fichier temporaire n'existe pas. Cela est dû à la manière dont l'agent de destination pour les transferts gérés utilise les fichiers temporaires lors de l'écriture d'un fichier de destination.

Utilisation des fichiers temporaires par un agent de destination

Par défaut, lorsqu'un transfert de fichiers géré est effectué, l'agent de destination effectue les étapes suivantes:

- Créez un fichier temporaire, appelé `destination_filename.part`.
- Verrouillez le fichier temporaire.
- Écrivez les données de fichier dans le fichier temporaire, lorsqu'il est reçu de l'agent source.

- Déverrouillez le fichier temporaire une fois que toutes les données du fichier ont été reçues et écrites.
- Renommez le fichier temporaire *destination_ filename .part* en *destination_ filename*.

Si un transfert géré fait l'objet d'une reprise, l'agent de destination peut créer des fichiers temporaires appelés *destination_ filename .partnumber*. L'agent de destination écrit ensuite les données du fichier dans ce fichier, au lieu de celui appelé *destination_ filename .part*.

Si le nom de fichier temporaire *destination_ filename .partnumber* existe déjà, l'agent de destination tente de créer un nouveau fichier temporaire nommé *destination_ filename .part(number + 1)*. Si ce fichier existe déjà, l'agent de destination tente de créer un fichier temporaire portant le nom *destination_ filename .part(number + 2)*, et ainsi de suite jusqu'à ce qu'il puisse créer le fichier. Dans le cas où l'agent tente, et échoue, de créer le fichier temporaire *destination_ filename .part1000*, il écrit directement dans le fichier de destination et n'utilise pas de fichier temporaire.

Lorsqu'un transfert géré est terminé, l'agent de destination supprime tous les fichiers temporaires appelés *destination_ filename .partnumber*, car il suppose qu'ils ont été créés par l'agent lors du transfert géré.

Remarque : Si la propriété d'agent **doNotUseTempOutputFile** est définie sur la valeur *true*, l'agent de destination n'utilise pas les fichiers temporaires. A la place, il écrit directement dans le fichier de destination. Pour plus d'informations sur la propriété **doNotUseTempOutputFile**, voir [Le fichier MFT agent.properties](#).

Pourquoi ce problème se produit

Une erreur BFGIO0341E est générée si l'agent de destination tente de renommer le fichier temporaire, uniquement pour trouver ce fichier qui n'existe plus. Un scénario typique qui peut provoquer ce problème est le suivant:

- Un *répertoire de transfert* a été configuré sur le système de fichiers cible.
- Un processus externe est configuré pour surveiller le *répertoire de transfert* et déplacer tous les fichiers qu'il trouve vers un nouvel emplacement.
- L'agent de destination crée et verrouille le fichier temporaire *destination_ filename .part* dans le *répertoire de transfert*.
- L'agent cible écrit les données de fichier dans le fichier temporaire.
- Une fois que toutes les données du fichier ont été écrites dans le fichier temporaire, l'agent de destination déverrouille le fichier.
- Le processus externe trouve le fichier temporaire et le déplace vers le nouvel emplacement.
- L'agent de destination tente de renommer le fichier temporaire et constate qu'il n'y est plus. Par conséquent, l'élément de transfert est marqué comme **Echec** avec une erreur BFGIO0341E.

Eviter ce problème

Il existe deux manières d'éviter que l'erreur BFGIO0341E ne se produise:

- Les fichiers temporaires écrits par un agent de destination se terminent toujours par le suffixe *.part* ou *.partnumber*. Si vous pouvez configurer le processus externe pour ignorer ces fichiers au lieu de les déplacer, les fichiers existeront toujours dans le répertoire cible lorsque l'agent de destination effectuera l'opération de changement de nom.
- Vous pouvez également configurer l'agent de destination de sorte qu'il n'utilise pas de fichiers temporaires et qu'il écrit directement dans le fichier de destination. Le fichier de destination est déverrouillé uniquement lorsque toutes les données du fichier y ont été écrites, ce qui permet au processus externe de le récupérer.

Pour configurer l'agent de destination pour qu'il écrive directement dans le fichier de destination, définissez la propriété d'agent **doNotUseTempOutputFile=true**. Pour plus d'informations sur cette propriété, voir [Le fichier MFT agent.properties](#).

Codes retour pour MFT

Les commandes Managed File Transfer , les tâches Ant et les messages de journal fournissent des codes retour indiquant si les fonctions ont abouti.

Le tableau suivant répertorie les codes retour de produit avec leur signification:

Code retour	Nom abrégé	Description
0	Réussite	La commande a abouti
1	Echec de la commande	L'exécution de la commande a échoué.
2	Délai d'attente de la commande dépassé	L'agent n'a pas répondu avec le statut de la commande dans un délai spécifié. Par défaut, ce délai d'attente est illimité pour les commandes d'appel géré et de transfert. Par exemple, lorsque vous spécifiez le paramètre -w avec la commande fteCreateTransfer . Par défaut, ce délai d'attente est de 5 secondes pour les autres commandes.
3	Délai d'accusé de réception dépassé	L'agent n'a pas accusé réception de la commande dans un délai spécifié. Par défaut, ce délai d'attente est de 5 secondes.
4	Agent incorrect	La commande a été envoyée à un agent incorrect. L'agent spécifié dans le XML de commande n'est pas l'agent qui lit la file d'attente de commandes dans laquelle le message a été placé.
20	Le transfert a partiellement abouti	Le transfert s'est terminé avec un succès partiel et certains fichiers ont été transférés.
21	Transfert arrêté	Le transfert a été arrêté par l'un des exits utilisateur.
22	Expiration du délai d'annulation du transfert	L'agent a reçu une demande d'annulation d'un transfert, mais l'annulation n'a pas pu être effectuée dans les 30 secondes. Le transfert n'a pas été annulé.

Tableau 24. Codes retour (suite)

Code retour	Nom abrégé	Description
26	ID d'annulation introuvable	L'agent a reçu une demande d'annulation d'un transfert, mais le transfert est introuvable. Le transfert a peut-être été exécuté avant que l'agent ne traite la demande d'annulation. Cela peut également être dû au fait que vous avez fourni un ID de transfert incorrect à la commande fteCancelTransfer . La demande d'annulation a été ignorée.
27	Annulation en cours	L'agent a reçu une demande d'annulation d'un transfert, mais celui-ci est déjà en cours d'annulation. La nouvelle demande d'annulation de transfert a été ignorée.
40	Echec	Le transfert a échoué et aucun des fichiers indiqués n'a été transféré.
41	Annulée	Le transfert a été annulé.
42	Echec du déclencheur	Le transfert n'a pas eu lieu parce que le transfert était conditionnel et que la condition requise n'était pas remplie.
43	XML syntaxiquement incorrect	Un message XML a été syntaxiquement incorrect.
44	Capacité de l'agent source dépassée	L'agent source ne disposait pas de la capacité suffisante pour effectuer le transfert.
45	Capacité de l'agent de destination dépassée	L'agent de destination ne disposait pas de la capacité suffisante pour effectuer le transfert.
46	Le nombre maximal de fichiers de l'agent source a été dépassé	Le nombre de fichiers transférés a dépassé la limite de l'agent source.
47	Le nombre maximal de fichiers de l'agent cible a été dépassé	Le nombre de fichiers transférés a dépassé la limite de l'agent cible.
48	Attributs de message de journal non valides	Un message de journal est syntaxiquement incorrect. Cette erreur est une erreur interne. Si vous recevez ce code retour, contactez le centre de support IBM pour obtenir de l'aide.

Tableau 24. Codes retour (suite)

Code retour	Nom abrégé	Description
49	Destination inaccessible	L'agent source ne peut pas envoyer de message à l'agent de destination en raison d'un problème IBM MQ . Par exemple, si le gestionnaire de files d'attente de l'agent source n'a pas été configuré correctement pour communiquer avec le gestionnaire de files d'attente de l'agent cible.
50	Violation de version d'essai	Une tentative a été effectuée par un agent de version d'évaluation pour communiquer avec un agent qui n'est pas un agent de version d'évaluation.
51	Transfert de source non autorisé	La propriété d'agent <code>maxSourceTransfers</code> a été définie sur 0. Il n'est pas permis que cet agent soit la source de tout transfert.
52	Transfert de destination non autorisé	La propriété d'agent <code>maxDestinationTransfers</code> a été définie sur 0. Il n'est pas autorisé que cet agent soit la destination des transferts.
53	Non autorisé	L'utilisateur n'est pas autorisé à effectuer l'opération. Pour plus de détails, consultez le message qui l'accompagne.
54	Les niveaux d'autorisation ne correspondent pas	La valeur de la propriété d'agent <code>authorityChecking</code> de l'agent source et de l'agent de destination ne correspondent pas.
55	Déclencheur non pris en charge	Une tentative a été effectuée pour créer un transfert avec un déclencheur sur un agent de pont de protocole. Ce comportement n'est pas pris en charge.
56	Le fichier de destination vers le message n'est pas pris en charge	L'agent de destination ne prend pas en charge l'écriture du fichier dans une file d'attente de destination
57	Espace fichier non pris en charge	L'agent cible ne prend pas en charge les espaces fichier.
58	Espace fichier rejeté	Le transfert d'espace fichier a été rejeté par l'agent de destination.

Tableau 24. Codes retour (suite)

Code retour	Nom abrégé	Description
59	Message de destination dans le fichier non pris en charge	L'agent de destination ne prend pas en charge les transferts de message à fichier.
64	Les deux files d'attente ne sont pas autorisées	La source et la destination d'un transfert sont une file d'attente.
65	Erreur générale de file d'attente de données	Une erreur s'est produite lors de l'accès à la file d'attente de données Managed File Transfer Agent .
66	Erreur d'autorisation d'insertion de file d'attente de données	Une erreur s'est produite lors de l'accès à la file d'attente de données Managed File Transfer Agent . Advanced Message Security n'est pas activé.
65	Erreur AMS d'insertion de file d'attente de données	Une erreur d'autorisation s'est produite lors de l'accès à la file d'attente de données Managed File Transfer Agent . Advanced Message Security est activée.
 69	Dépassement du délai de récupération du transfert	La récupération d'un transfert a expiré après la valeur de délai d'attente transferRecoveryspécifiée.
70	L'agent s'est arrêté de manière anormale	L'application a rencontré un problème irrémédiable et est en train de s'arrêter de force.
75	Le gestionnaire de files d'attente est indisponible	L'application ne peut pas continuer car le gestionnaire de files d'attente de l'application n'est pas disponible.
78	Problème lié à la configuration de démarrage	L'application ne peut pas continuer en raison d'un problème lié aux données de configuration de démarrage.
85	Problème avec le serveur de base de données	L'application ne peut pas continuer car il y a un problème avec la base de données (généralement renvoyé uniquement par un consignateur)
100	Substitution de moniteur incorrecte	Le format d'une substitution de variable dans un script XML de tâche de contrôle est incorrect.
101	Ressource de moniteur incorrecte	Le nombre de définitions de ressources de moniteur n'était pas valide.

Tableau 24. Codes retour (suite)

Code retour	Nom abrégé	Description
102	Déclencheur de moniteur incorrect	Le nombre de définitions de déclencheur de moniteur n'était pas valide.
103	Tâche de surveillance incorrecte	Le nombre de définitions de tâche de contrôle n'était pas valide.
104	Moniteur manquant	Le moniteur demandé n'est pas présent.
105	Moniteur déjà présent	Le moniteur demandé est déjà présent.
106	Erreur d'exit utilisateur de surveillance	Un exit utilisateur de moniteur a généré une erreur lors d'une interrogation du moniteur de ressources.
107	Exit utilisateur de surveillance annulé	Un exit utilisateur de surveillance a demandé l'annulation d'une transaction.
108	Echec de la tâche de surveillance	Une tâche de surveillance n'a pas pu se terminer en raison d'une erreur lors du traitement de la tâche.
109	Echec de la surveillance de la ressource	Une définition de ressource de moniteur ne peut pas être appliquée à la ressource donnée.
110	Echec de la substitution des variables de la tâche de surveillance	Une variable a été spécifiée dans une tâche de contrôle mais aucun nom correspondant n'a été trouvé dans les métadonnées. Par conséquent, la variable ne peut pas être remplacée par une valeur.
111	L'agent source de la tâche de surveillance n'est pas valide	L'agent source de la tâche de transfert du moniteur ne correspond pas à l'agent du moniteur de ressources.
112	Gestionnaire de files d'attente source de la tâche de surveillance non valide	Le gestionnaire de files d'attente de l'agent source de la tâche de transfert du moniteur ne correspond pas au gestionnaire de files d'attente de l'agent du moniteur de ressources.

Tableau 24. Codes retour (suite)

Code retour	Nom abrégé	Description
113	Moniteur non pris en charge	Une tentative de création ou de suppression d'un moniteur de ressources a été effectuée sur un agent de pont de protocole. Ce comportement n'est pas pris en charge.
114	Ressource de moniteur refusée	L'accès est refusé au répertoire analysé par la ressource du moniteur.
115	File d'attente de ressources de surveillance en cours d'utilisation	La file d'attente de ressources du moniteur est déjà ouverte et n'est pas compatible avec l'entrée avec l'accès partagé.
116	File d'attente de ressources du moniteur inconnue	La file d'attente de ressources du moniteur n'existe pas sur le gestionnaire de files d'attente associé du moniteur.
118	Expression de ressource de contrôle non valide	Une erreur s'est produite lors de l'évaluation de l'expression XPath. L'expression XPath est évaluée pour accéder aux propriétés définies par l'utilisateur dans l'en-tête du message. Le message se trouve dans une file d'attente qui est contrôlée par le moniteur de ressources.
119	Gestionnaire de files d'attente de l'agent source de la tâche de surveillance manquant	Le nom de l'agent source ou le nom du gestionnaire de files d'attente de l'agent source est manquant dans la définition de la tâche de surveillance.
130	File d'attente de surveillance non activée	La file d'attente de ressources du moniteur n'est pas activée.
121	Erreur inattendue lors de l'accès à la file d'attente du moniteur	Une erreur inattendue s'est produite lors de l'accès à la file d'attente de ressources du moniteur.
122	File d'attente de commandes de contrôle non activée pour l'ID de contexte	La file d'attente de commandes de l'agent de contrôle n'est pas activée pour l'identification de contexte définie.

Le tableau suivant répertorie les codes de réponse intermédiaires du produit avec leur signification:

Code de réponse	Nom abrégé	Description
-2	Accusé de réception	La demande a été reçue mais elle est en attente d'achèvement.
-3	PROGRESS	La demande concerne un certain nombre de fichiers et certains sont toujours en attente d'achèvement.

Remarque :

Les codes de réponse ne sont présents que si le processus qui génère la demande fournit une file d'attente de réponse. Il s'agit de réponses intermédiaires et les commandes Managed File Transfer renvoient uniquement le code de réponse final.

Référence associée

«Codes retour pour les fichiers d'un transfert», à la page 353

Les fichiers individuels d'un transfert ont leurs propres codes de résultat qui ont des significations différentes du code retour global d'une commande.

Codes retour pour les fichiers d'un transfert

Les fichiers individuels d'un transfert ont leurs propres codes de résultat qui ont des significations différentes du code retour global d'une commande.

Dans un message de progression du journal de transfert comportant un élément <action> défini sur la valeur "progress", chaque fichier signalé comporte un élément <status> avec un resultCode.

Exemple :

```
<action time="2009-11-23T21:28:09.593Z">progress</action>

...
  <status resultCode="1">
    <supplement>BFGI00006E: File "C:\destinationfiles\dest1.doc"
      already exists.</supplement>
  </status>
```

Le tableau suivant décrit les valeurs possibles pour resultCode:

Valeur de code de résultat	Description
0	Aboutissement Le fichier a été transféré.
1	Echec. Le transfert du fichier a échoué. Voir l'élément < supplement> pour plus de détails sur l'erreur.
2	Avertissement. Le fichier a été transféré mais un message d'avertissement a été signalé. Par exemple, le fichier source ne peut pas être supprimé bien que la disposition de la source soit définie pour la suppression. Voir l'élément < supplement> pour plus de détails sur l'avertissement.

Traitement des incidents liés au pont Connect:Direct

Utilisez les informations de référence et les exemples suivants pour vous aider à diagnostiquer les erreurs renvoyées par le pont Connect:Direct .

- [«Traçage du pont Connect:Direct», à la page 354](#)
- [«Informations de journal pour le pont Connect:Direct», à la page 354](#)
- [«Résolution des problèmes de droits d'accès avec les noeuds Connect:Direct», à la page 355](#)
- [«Que faire si les transferts de texte vers ou depuis des noeuds Connect:Direct ne convertissent pas correctement les données», à la page 356](#)
-  [«Que faire si les transferts vers des membres PDS ou PDS via le pont Connect:Direct échouent», à la page 356](#)
-  [«Chemins d'accès aux fichiers Connect:Direct spécifiés avec une double barre oblique», à la page 357](#)
- [«Augmentation du nombre de transferts simultanés pour le pont Connect:Direct», à la page 357](#)
- [«Débogage d'un processus Connect:Direct appelé par un transfert de fichier», à la page 358](#)

Traçage du pont Connect:Direct

Vous pouvez capturer la trace à partir du noeud Connect:Direct qui fait partie du pont Connect:Direct pour faciliter l'identification des problèmes.

Pourquoi et quand exécuter cette tâche

Pour activer la trace, procédez comme suit:

Procédure

1. Arrêtez l'agent de pont Connect:Direct .
2. Editez le fichier de propriétés de l'agent de pont Connect:Direct pour inclure la ligne suivante:

```
cdTrace=true
```

3. Démarrez l'agent de pont Connect:Direct .

Résultats

Les informations de trace sont écrites dans le fichier `output0.log` du répertoire de configuration de l'agent de pont Connect:Direct .

Référence associée

[«Le fichier MFT agent.properties», à la page 564](#)

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.

Informations de journal pour le pont Connect:Direct

Vous pouvez utiliser un agent de pont Connect:Direct pour transférer des fichiers entre des agents MFT et des noeuds Connect:Direct . Les informations de journal sur les noeuds et processus Connect:Direct impliqués dans ces transferts sont affichées dans le plug-in IBM MQ Explorer et sont stockées dans votre base de données de journaux.

L'agent de pont Connect:Direct doit être IBM WebSphere MQ File Transfer Edition 7.0.4 ou une version ultérieure. L'autre agent impliqué dans le transfert peut être n'importe quelle version de Managed File Transfer. Toutefois, pour que des informations soient consignées sur les noeuds et les processus Connect:Direct , tous les agents MFT impliqués dans le transfert doivent être IBM WebSphere MQ File Transfer Edition 7.0.4 ou une version ultérieure. Pour que ces informations s'affichent dans le plug-in IBM MQ Explorer , le plug-in doit être IBM WebSphere MQ File Transfer Edition 7.0.4 ou version ultérieure. Pour que ces informations soient stockées dans la base de données de journaux, le consignateur de base

de données et le schéma de base de données doivent être IBM WebSphere MQ File Transfer Edition 7.0.4 ou une version ultérieure.

Les informations de journal sur les noeuds Connect:Direct et les processus Connect:Direct impliqués dans un transfert de fichier sont incluses dans les messages de journal qui sont publiés dans SYSTEM.FTE sur le gestionnaire de file d'attente de coordination. Pour plus d'informations, voir la section «[Formats des messages du journal de transfert de fichiers](#)», à la page 771.

Les informations suivantes sont incluses dans le message publié:

- Nom de noeud de pont Connect:Direct
- Nom du noeud principal (PNODE)
- Nom du noeud secondaire (SNODE)
- Nom du processus
- Numéro d'ID de processus

Le noeud de pont Connect:Direct est le même que le noeud principal ou le noeud secondaire.

La valeur du nom de noeud de pont Connect:Direct correspond au nom sous lequel le noeud de pont est connu de l'agent de pont MFT Connect:Direct . Les noms de noeud principal et secondaire sont les noms utilisés pour faire référence aux noeuds de la mappe de réseau du noeud de pont Connect:Direct .

Référence associée

«[Exemples de message de journal de transfert de pont Connect:Direct](#)», à la page 790

L'élément `destinationAgent` ou `sourceAgent` contient des attributs supplémentaires lorsque l'agent cible ou l'agent source est un agent de pont Connect:Direct . Le message de journal Démarré ne contient qu'un sous-ensemble des informations sur le transfert Connect:Direct . Les messages de journal Progression et Terminé contiennent des informations complètes sur le transfert Connect:Direct .

Résolution des problèmes de droits d'accès avec les noeuds Connect:Direct

Utilisez les informations de cette rubrique si vos transferts entre Managed File Transfer et Connect:Direct échouent avec une erreur indiquant que les droits sont insuffisants.

Pour les transferts impliquant le pont Connect:Direct , l'ID utilisateur qui se connecte au noeud Connect:Direct est déterminé par l'ID utilisateur IBM MQ Message Descriptor (MQMD) associé à la demande de transfert. Vous pouvez mapper des ID utilisateur MQMD spécifiques à des ID utilisateur Connect:Direct spécifiques. Pour plus d'informations, voir la section «[Mappage des données d'identification pour Connect:Direct](#)», à la page 149.

Les transferts peuvent échouer avec l'une des erreurs suivantes:

- ```
BFGCD0001E: This task was rejected by the Connect:Direct API with the following error message: Connect:Direct Node detected error.
LCCA000I The user has no functional authority to issue the selp command
```
- ```
BFGCD0026I: Connect:Direct messages: The submit of the process succeeded. Process number 1092 (name F35079AE, SNODE MYNODE) executing. User fteuser does not have permission to override SNODEID. User fteuser does not have permission to override SNODEID. User fteuser does not have permission to override SNODEID.
```

Si vous voyez l'une de ces erreurs, déterminez quel ID utilisateur Connect:Direct est associé à l'ID utilisateur MQMD qui a été utilisé pour la demande de transfert. Cet ID utilisateur Connect:Direct doit disposer des droits permettant d'effectuer les opérations Connect:Direct requises par le pont Connect:Direct . Pour obtenir la liste des droits fonctionnels nécessaires et des conseils sur la façon de les accorder, voir «[Mappage des données d'identification pour Connect:Direct à l'aide du fichier ConnectDirectCredentials.xml](#)», à la page 149.

Que faire si les transferts de texte vers ou depuis des noeuds Connect:Direct ne convertissent pas correctement les données

Lorsque vous transférez des fichiers en mode texte entre un agent MFT et un noeud Connect:Direct, la page de codes et la conversion de caractères de fin de ligne sont effectuées. Le transfert utilise les informations du système d'exploitation dans la mappe de réseau du noeud de pont Connect:Direct pour déterminer les caractères de fin de ligne d'un noeud distant. Si les informations de la mappe de réseau sont incorrectes, la conversion de caractères de fin de ligne peut être effectuée de manière incorrecte.

Vérifiez que la mappe de réseau du noeud de pont Connect:Direct et tous les noeuds Connect:Direct utilisés comme destination de transfert incluent la description de plateforme appropriée.

- Si votre noeud de pont Connect:Direct se trouve sur un système Windows, assurez-vous de sélectionner la valeur correcte dans la liste **Système d'exploitation** pour chaque noeud distant.
 - Si le noeud distant se trouve sur un système Windows, sélectionnez Windows.
 - Si le noeud distant se trouve sur un système UNIX ou Linux, sélectionnez UNIX.
 -  Si le noeud distant se trouve sur un système z/OS, sélectionnez OS/390.

Les transferts sur des noeuds distants sur d'autres systèmes d'exploitation ne sont pas pris en charge par le pont Connect:Direct.

- Vérifiez que, pour chaque noeud à distance vers ou depuis lequel vous transférez un fichier, indiquez le type de système d'exploitation du noeud Connect:Direct à distance dans le fichier `ConnectDirectNodeProperties.xml` du répertoire de configuration de l'agent de pont Connect:Direct. Pour plus d'informations, voir «[Configurez le fichier ConnectDirectNodeProperties.xml pour inclure des informations sur les noeuds Connect:Direct distants](#)», à la page 148 et «[Format du fichier de propriétés du noeud Connect:Direct](#)», à la page 603.

Référence associée

«Transfert de fichiers texte entre Connect:Direct et MFT», à la page 665

Le transfert de texte implique la conversion de la page de codes d'un fichier d'une page de codes à une autre. Le transfert de fichiers texte implique également la conversion de caractères CRLF (saut de ligne de retour chariot) entre les systèmes. Cette rubrique récapitule le comportement du transfert de fichiers texte dans les transferts entre un Managed File Transfer Agent et un noeud Connect:Direct.

Que faire si les transferts vers des membres PDS ou PDS via le pont Connect:Direct échouent

Si la destination d'un transfert est un noeud Connect:Direct sur z/OS et qu'il s'agit d'un membre PDS ou PDS, le transfert échoue si le paramètre **-de** n'a pas été spécifié avec la valeur `overwrite`.

Pourquoi et quand exécuter cette tâche

Si vous avez soumis le transfert à l'aide de la commande **fteCreateTransfer** ou **fteCreateTemplate**, procédez comme suit:

Procédure

1. Modifiez la commande que vous avez soumise pour inclure **-de** `overwrite`.
2. Réexécutez la commande.

Utilisation du plug-in IBM MQ Explorer

Pourquoi et quand exécuter cette tâche

Si vous avez soumis le transfert à l'aide du plug-in IBM MQ Explorer, procédez comme suit:

Procédure

1. Spécifiez les informations source et de destination dans l'assistant **Créer un transfert de fichier géré**.
2. Sélectionnez **Ecraser les fichiers du système de fichiers cible portant le même nom**.
3. Réexécutez la commande.

Chemins d'accès aux fichiers Connect:Direct spécifiés avec une double barre oblique

Si, dans le cadre d'un transfert de fichier, vous spécifiez un fichier situé sur un noeud Connect:Direct en utilisant un chemin de fichier commençant par une double barre oblique (/), le fichier est traité comme un jeu de données.

Les sources et les destinations sur un noeud Connect:Direct sont spécifiées au format `cd_node_name:file_path`. Si `file_path` commence par une double barre oblique (/), la source ou la destination est traitée comme un fichier. C'est le cas même lorsque le noeud Connect:Direct n'est pas sur z/OS. Cela peut entraîner des échecs de transfert si le chemin d'accès au fichier est accidentellement spécifié avec une double barre oblique (/) au début et que le fichier n'est pas un jeu de données.

Veillez à ne pas spécifier un `file_path` commençant par une double barre oblique (/) si vous ne souhaitez pas que le fichier que vous spécifiez soit traité comme un jeu de données.

Concepts associés

«Traitement des incidents liés au pont Connect:Direct», à la page 353

Utilisez les informations de référence et les exemples suivants pour vous aider à diagnostiquer les erreurs renvoyées par le pont Connect:Direct.

Référence associée

«Transfert de fichiers vers et depuis des noeuds Connect:Direct», à la page 653

Vous pouvez transférer des ensembles de données entre les agents Managed File Transfer et les noeuds IBM Sterling Connect:Direct à l'aide du pont Connect:Direct. Vous pouvez spécifier un fichier comme source de transfert, destination de transfert ou les deux.

Augmentation du nombre de transferts simultanés pour le pont Connect:Direct

Pour augmenter le nombre de transferts simultanés que l'agent de pont Connect:Direct peut traiter, vous devez modifier trois propriétés d'agent. Vous devez également augmenter le nombre maximal de connexions acceptées par le noeud Connect:Direct.

Le nombre maximal de transferts simultanés qu'un agent de pont Connect:Direct peut traiter dépend des valeurs de certaines propriétés d'agent. Les propriétés d'agent **maxSourceTransfers** et **maxDestinationTransfers** ont une valeur par défaut de cinq transferts pour un agent de pont Connect:Direct. Cette valeur par défaut est inférieure à la valeur par défaut de 25 transferts pour les autres types d'agent. Un pont Connect:Direct, où l'agent est configuré avec les valeurs par défaut **maxSourceTransfers** et **maxDestinationTransfers**, peut traiter jusqu'à 10 transferts à la fois: cinq transferts où l'agent est la source et cinq transferts où l'agent est la destination.

Ces valeurs par défaut garantissent que l'agent de pont Connect:Direct ne dépasse pas le nombre maximal de connexions d'API au noeud Connect:Direct. Un agent de pont Connect:Direct avec la configuration par défaut utilise un maximum de 10 connexions d'API au noeud Connect:Direct. Le nombre maximal de connexions acceptées par un noeud Connect:Direct sur UNIX est contrôlé par le paramètre **api.max.connects** Connect:Direct. Pour un noeud Connect:Direct sous Windows, le paramètre équivalent est **max.api.connects**.

Si le débit auquel votre pont Connect:Direct effectue un grand nombre de transferts de fichiers n'est pas suffisant, vous pouvez augmenter le nombre de transferts simultanés traités par l'agent de pont Connect:Direct. Modifiez les propriétés d'agent suivantes pour l'agent de pont Connect:Direct :

maxSourceTransfers

Définissez cette propriété sur une valeur supérieure à 5, mais inférieure ou égale à 25. Si vous choisissez une valeur supérieure à 25, l'agent risque d'être à court de mémoire, sauf si vous augmentez la quantité de mémoire disponible pour la machine virtuelle Java utilisée par l'agent.

maxDestinationTransfers

Définissez cette propriété sur une valeur supérieure à 5, mais inférieure ou égale à 25. Si vous choisissez une valeur supérieure à 25, l'agent risque d'être à court de mémoire, sauf si vous augmentez la quantité de mémoire disponible pour la machine virtuelle Java utilisée par l'agent.

ioThreadPoolSize

La valeur par défaut de **ioThreadPoolSize** est 10. Cette propriété limite le nombre de connexions d'API de noeud Connect:Direct pour les transferts où l'agent de pont Connect:Direct est l'agent source. Ces transferts sont de Connect:Direct vers Managed File Transfer. Utilisez les instructions suivantes pour définir la valeur de cette propriété:

- Si la valeur de **maxSourceTransfers** est inférieure à la valeur de **maxDestinationTransfers**, définissez **ioThreadPoolSize** pour doubler la valeur de **maxSourceTransfers** ou de 10, la valeur la plus élevée étant retenue.
- Si la valeur de **maxSourceTransfers** est supérieure à la valeur de **maxDestinationTransfers**, définissez **ioThreadPoolSize** sur la somme de **maxSourceTransfers** et de **maxDestinationTransfers**

En plus de ces propriétés d'agent, vous devez également modifier le nombre maximal de connexions d'API simultanées pour le noeud Connect:Direct qui fait partie du pont Connect:Direct . Le paramètre Connect:Direct qui contrôle ce nombre est **api.max.connects** si votre noeud est sous UNIX, ou **max.api.connects** si votre noeud est sous Windows. Apportez les modifications suivantes au paramètre approprié:

api.max.connects (si le noeud de votre pont Connect:Direct est sous UNIX)

Définissez ce paramètre sur une valeur supérieure à la somme de **maxSourceTransfers** et **maxDestinationTransfers**. La valeur par défaut du paramètre **api.max.connects** est 16. Pour plus d'informations sur la définition de ce paramètre, voir la documentation Connect:Direct .

max.api.connects (si le noeud de votre pont Connect:Direct est sous Windows)

Définissez ce paramètre sur une valeur supérieure à la somme de **maxSourceTransfers** et **maxDestinationTransfers**. La valeur par défaut du paramètre **max.api.connects** est 10. Pour plus d'informations sur la définition de ce paramètre, voir la documentation Connect:Direct .

Tâches associées

«[Configuration du pont Connect:Direct](#)», à la page 146

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

«[Le fichier MFT agent.properties](#)», à la page 564

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.

Débogage d'un processus Connect:Direct appelé par un transfert de fichier

Vous pouvez configurer l'agent de pont Connect:Direct pour écrire des informations de journal sur le processus Connect:Direct appelé par un transfert de fichier dans le fichier `output0.log` du répertoire de configuration de l'agent de pont Connect:Direct .

Pourquoi et quand exécuter cette tâche

Pour configurer la consignation des processus Connect:Direct , procédez comme suit:

Procédure

1. Arrêtez l'agent de pont Connect:Direct .
2. Editez le fichier `agent.properties` dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name` pour inclure la propriété `logCDProcess`.
La propriété `logCDProcess` peut avoir l'une des valeurs suivantes:
 - Aucun-Aucune information n'est consignée. Il s'agit de l'option par défaut.
 - Echecs-Des informations sur les processus Connect:Direct ayant échoué sont consignées.
 - Toutes-Les informations sur tous les processus Connect:Direct sont consignées.
3. Démarrez l'agent de pont Connect:Direct .

Résultats

Les informations sur les processus Connect:Direct sont consignées dans le fichier `output0.log` de l'agent de pont Connect:Direct . Les informations consignées comprennent:

- MFT ID de transfert
- Connect:Direct Nom de processus
- Connect:Direct Numéro de processus
- Définition de processus générée
- Nom de fichier du modèle de processus, si le processus Connect:Direct est défini par l'utilisateur

Concepts associés

«[Traitement des incidents liés au pont Connect:Direct](#)», à la page 353

Utilisez les informations de référence et les exemples suivants pour vous aider à diagnostiquer les erreurs renvoyées par le pont Connect:Direct .

Référence associée

«[Le fichier MFT agent.properties](#)», à la page 564

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.

Référence MFT

Référence de sécurité MFT

Droits d'accès au système de fichiers pour MFT dans IBM MQ

Lorsque vous installez et configurez le composant Managed File Transfer de IBM MQ, les répertoires de configuration, d'installation et de logs sont créés avec les droits suivants.

UNIX et Linux



Tableau 27. Récapitulatif des droits d'accès aux répertoires sous UNIX et Linux

Répertoire	Droits d'accès
/var/mqm/mqft/config	<ul style="list-style-type: none"> • Accessible en écriture par le groupe mqm • Lisible par tous <p>Les utilisateurs du groupe mqm ont accès en écriture à ces répertoires et fichiers</p>
/var/mqm/mqft/installations	<ul style="list-style-type: none"> • Accessible en écriture par le groupe mqm • Lisible par tous
/var/mqm/mqft/logs	Accessible en lecture et en écriture

Windows

Windows

Tableau 28. Récapitulatif des droits d'accès aux répertoires sous Windows

Répertoire	Droits d'accès
MQ_DATA_PATH\mqft\config	<p>Les utilisateurs suivants disposent d'un accès complet en lecture et en écriture:</p> <ul style="list-style-type: none"> • Administrateurs • Compte système • groupe mqm <p>D'autres utilisateurs disposent d'un accès en lecture</p>
MQ_DATA_PATH\mqft\installations	<p>Les utilisateurs suivants disposent d'un accès complet en lecture et en écriture:</p> <ul style="list-style-type: none"> • Administrateurs • Compte système • groupe mqm <p>D'autres utilisateurs disposent d'un accès en lecture</p>
MQ_DATA_PATH\mqft\logs	<p>Les utilisateurs suivants disposent d'un accès complet en lecture et en écriture:</p> <ul style="list-style-type: none"> • Administrateurs • Compte système • groupe mqm <p>Les autres utilisateurs disposent d'un accès en lecture et en écriture</p>

z/OS

z/OS

Tableau 29. Récapitulatif des droits d'accès aux répertoires sous z/OS

Répertoire	Droits d'accès
DATA_PATH/mqft/config	<ul style="list-style-type: none"> • Accessible en écriture par le groupe mqm ou par le nom de groupe identifié dans la variable d'environnement BFG_GROUP_NAME • Lisible par tous <p>Les utilisateurs du groupe mqm, ou la valeur de la variable d'environnement BFG_GROUP_NAME, ont un accès en écriture à ces répertoires et fichiers</p>
DATA_PATH/mqft/installations	<ul style="list-style-type: none"> • Accessible en écriture par le groupe mqm ou par le nom de groupe identifié dans la variable d'environnement BFG_GROUP_NAME • Lisible par tous
DATA_PATH/mqft/logs	Accessible en lecture et en écriture

Remarque : DATA_PATH est dérivé de la variable d'environnement BFG_DATA.

Droits de gestion pour les ressources spécifiques à MFT

Pour toute demande de transfert de fichier, les processus de l'agent Managed File Transfer requièrent un certain niveau d'accès à leurs systèmes de fichiers locaux. En outre, l'ID utilisateur associé au processus d'agent et les ID utilisateur associés aux utilisateurs effectuant des opérations de transfert de fichiers doivent être autorisés à utiliser certains objets IBM MQ .

Les commandes sont émises par les utilisateurs, qui peuvent avoir un rôle opérationnel où ils démarrent généralement un transfert de fichier. Ils peuvent également avoir un rôle d'administration dans lequel ils peuvent également contrôler le moment où les agents sont créés, démarrés, supprimés ou nettoyés (c'est-à-dire, lorsque les messages de toutes les files d'attente système de l'agent sont supprimés). Les messages contenant des demandes de commande sont placés sur le système SYSTEM.FTE.COMMAND COMMAND lorsqu'un utilisateur émet une commande. Le processus d'agent extrait les messages contenant des demandes de commande de SYSTEM.FTE.COMMAND . Le processus d'agent utilise également quatre autres files d'attente système, qui sont les suivantes:

- SYSTEME SYSTEM.FTE.DATA.nom_agent
- SYSTEME SYSTEM.FTE.EVENT.nom_agent
- SYSTEME SYSTEM.FTE.REPLY.nom_agent
- SYSTEME SYSTEM.FTE.STATE.nom_agent

Etant donné que les utilisateurs émettant des commandes utilisent les files d'attente répertoriées précédemment de différentes manières pour le processus d'agent, affectez des droits IBM MQ différents aux ID utilisateur ou aux groupes d'utilisateurs associés à chacun d'eux. Pour plus d'informations, voir [«Restriction des droits de groupe pour les ressources spécifiques à MFT»](#), à la page 362.

L'agent comporte des files d'attente supplémentaires qui peuvent être utilisées pour accorder aux utilisateurs le droit d'effectuer certaines actions. Pour plus d'informations sur l'utilisation des files d'attente de droits d'accès, voir [«Restriction des droits utilisateur sur les actions de l'agent MFT»](#), à la page 368 . L'agent ne place ni n'obtient de messages dans ces files d'attente. Toutefois, vous devez vous assurer que les files d'attente disposent des droits IBM MQ appropriés pour l'ID utilisateur utilisé pour exécuter le processus d'agent, ainsi que pour les ID utilisateur associés aux utilisateurs qui sont autorisés à effectuer certaines actions. Les files d'attente de droits d'accès sont les suivantes:

- 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*

Si vous effectuez une migration depuis une version de Managed File Transfer antérieure à 7.0.2 vers IBM WebSphere MQ 7.5 ou une version ultérieure et que vous conservez des configurations d'agent existantes, vous devez créer les files d'attente de droits d'accès manuellement. Utilisez la commande MQSC suivante pour créer les files d'attente:

```
DEFINE QLOCAL(authority_queue_name) DEFPRTY(0) DEFSOPT(SHARED) GET(ENABLED) MAXDEPTH(0) +
  MAXMSGL(0) MSGDLVSQ(PRIORITY) PUT(ENABLED) RETINTVL(99999999) SHARE NOTRIGGER +
  USAGE(NORMAL) REPLACE
```

Le processus d'agent publie également des messages dans SYSTEM.FTE sur le gestionnaire de file d'attente de coordination à l'aide de SYSTEM.FTE, file d'attente. Selon que le processus d'agent a le rôle d'agent source ou de destination, le processus d'agent peut avoir besoin de droits de lecture, d'écriture, de mise à jour et de suppression des fichiers.

Vous pouvez créer et modifier des enregistrements de droits d'accès pour des objets IBM MQ à l'aide de IBM MQ Explorer. Cliquez avec le bouton droit de la souris sur l'objet, puis cliquez sur **Droits sur les objets > Gérer les enregistrements de droits d'accès**. Vous pouvez également créer des enregistrements de droits d'accès à l'aide de la commande **setmqaut**, décrite dans la rubrique [Commande setmqaut \(grant or revoke authority\)](#).

Référence associée

«Restriction des droits de groupe pour les ressources spécifiques à MFT», à la page 362

Au lieu d'accorder des droits à des utilisateurs individuels pour tous les différents objets pouvant être impliqués, configurez deux groupes de sécurité pour l'administration du contrôle d'accès Managed File Transfer : FTEUSER et FTEAGENT. Il incombe à l'administrateur IBM MQ de créer et de remplir ces groupes. L'administrateur peut choisir d'étendre ou de modifier la configuration proposée décrite ici.

«Restriction des droits utilisateur sur les actions de l'agent MFT», à la page 368

En plus d'utiliser des groupes pour gérer l'accès aux ressources, vous pouvez activer un niveau supplémentaire de sécurité afin de restreindre les actions d'agent Managed File Transfer qu'un utilisateur peut effectuer. Accordez des autorisations à un utilisateur dans une file d'attente d'autorisations d'agent pour permettre à l'utilisateur d'effectuer des actions d'agent spécifiques.

«Droits d'accès au consignateur MFT», à la page 372

L'utilisateur du système d'exploitation qui exécute le consignateur requiert certains droits IBM MQ sur les files d'attente du consignateur et sur SYSTEM.FTE.

Restriction des droits de groupe pour les ressources spécifiques à MFT

Au lieu d'accorder des droits à des utilisateurs individuels pour tous les différents objets pouvant être impliqués, configurez deux groupes de sécurité pour l'administration du contrôle d'accès Managed File Transfer : FTEUSER et FTEAGENT. Il incombe à l'administrateur IBM MQ de créer et de remplir ces groupes. L'administrateur peut choisir d'étendre ou de modifier la configuration proposée décrite ici.

Droits de connexion aux gestionnaires de files d'attente

Les commandes exécutées par les utilisateurs opérationnels, les administrateurs et IBM MQ Explorer doivent pouvoir se connecter au gestionnaire de files d'attente de commandes et au gestionnaire de files d'attente de coordination. Le processus d'agent et les commandes exécutées pour créer, modifier ou supprimer l'agent doivent pouvoir se connecter au gestionnaire de files d'attente d'agent.

- Accordez au groupe FTEUSER le droit de connexion pour le gestionnaire de files d'attente de commandes et le gestionnaire de files d'attente de coordination. Exemple :

ULW Pour les systèmes UNIX, Linux et Windows :

```
setmqaut -m command_queue_manager -t qmgr -g FTEUSER +connect  
setmqaut -m coordination_queue_manager -t qmgr -g FTEUSER +connect
```

IBM i Pour IBM i :

```
GRTRMQAUT OBJ('command_queue_manager') OBJTYPE(*MQM) USER(FTEUSER) AUT(*CONNECT)  
GRTRMQAUT OBJ('coordination_queue_manager') OBJTYPE(*MQM) USER(FTEUSER) AUT(*CONNECT)
```

z/OS Pour z/OS :

```
RDEFINE MQCONN command_queue_manager.BATCH UACC(NONE)  
PERMIT command_queue_manager.BATCH CLASS(MQCONN) ID(FTEUSER) ACCESS(READ)  
RDEFINE MQCONN coordination_queue_manager.BATCH UACC(NONE)  
PERMIT coordination_queue_manager.BATCH CLASS(MQCONN) ID(FTEUSER) ACCESS(READ)
```

- Accordez au groupe FTEAGENT le droit de connexion et d'interrogation sur le gestionnaire de files d'attente de l'agent. Exemple :

ULW Pour les systèmes UNIX, Linux et Windows :

```
setmqaut -m agent_queue_manager -t qmgr -g FTEAGENT +connect +inq +setid
```

IBM i Pour IBM i :

```
GRTRMQAUT OBJ('agent_queue_manager') OBJTYPE(*MQM) USER(FTEAGENT) AUT(*CONNECT)
```

z/OS Pour z/OS :

```
RDEFINE MQCONN agent_queue_manager.BATCH UACC(NONE)  
PERMIT agent_queue_manager.BATCH CLASS(MQCONN) ID(FTEAGENT) ACCESS(READ)
```

Pour plus d'informations sur les commandes qui se connectent directement à un gestionnaire de files d'attente, voir [«Les commandes et processus MFT qui se connectent à quel gestionnaire de files d'attente»](#), à la page 376

Droit d'insertion d'un message dans la file d'attente COMMAND appartenant à l'agent

La file d'attente de commandes de l'agent doit être disponible pour tout utilisateur autorisé à demander que l'agent effectue une action. Pour satisfaire à cette exigence,

- Accordez au groupe FTEUSER uniquement l'accès en insertion à SYSTEM.FTE.COMMAND.*nom_agent* queue. Exemple :

ULW Pour les systèmes UNIX, Linux et Windows :

```
setmqaut -m QM1 -n SYSTEM.FTE.COMMAND.agent_name -t queue -g FTEUSER +put
```

IBM i Pour IBM i :

```
GRTRMQAUT OBJ('SYSTEM.FTE.COMMAND.agent_name') OBJTYPE(*Q) USER(FTEUSER) AUT(*PUT)  
MQMNAME('QM1')
```

z/OS Pour z/OS :

```
RDEFINE MQQUEUE QM1.SYSTEM.FTE.COMMAND.agent_name UACC(NONE)  
PERMIT QM1.SYSTEM.FTE.COMMAND.agent_name CLASS(MQQUEUE) ID(FTEUSER) ACCESS(UPDATE)
```

- Accordez au groupe FTEAGENT les droits d'accès (put, get et setid) au système SYSTEM.FTE.COMMAND.*nom_agent* queue. Exemple :

ULW**Pour les systèmes UNIX, Linux et Windows :**

```
setmqaut -m QM1 -n SYSTEM.FTE.COMMAND.agent_name -t queue -g FTEAGENT +browse +put +get +setid
```

IBM i**Pour IBM i :**

```
GRTMQMAUT OBJ('SYSTEM.FTE.COMMAND.agent_name') OBJTYPE(*Q) USER(FTEAGENT) AUT(*PUT)
MQMNAME('QM1')
GRTMQMAUT OBJ('SYSTEM.FTE.COMMAND.agent_name') OBJTYPE(*Q) USER(FTEAGENT) AUT(*GET)
MQMNAME('QM1')
GRTMQMAUT OBJ('SYSTEM.FTE.COMMAND.agent_name') OBJTYPE(*Q) USER(FTEAGENT) AUT(*SETID)
MQMNAME('QM1')
```

z/OS**Pour z/OS :**

```
RDEFINE MQQUEUE QM1.SYSTEM.FTE.COMMAND.agent_name UACC(NONE)
PERMIT QM1.SYSTEM.FTE.COMMAND.agent_name CLASS(MQQUEUE) ID(FTEAGENT) ACCESS(UPDATE)
RDEFINE MQADMIN QM1.CONTEXT.SYSTEM.FTE.COMMAND.agent_name UACC(NONE)
PERMIT QM1.CONTEXT.SYSTEM.FTE.COMMAND.agent_name CLASS(MQADMIN) ID(FTEAGENT)
ACCESS(UPDATE)
```

Les agents doivent avoir accès aux messages placés dans les files d'attente de commandes des autres agents. Si des agents sont connectés à des gestionnaires de files d'attente éloignées, vous devrez peut-être accorder des autorisations supplémentaires pour permettre au canal d'insérer des messages dans cette file d'attente.

Droit d'insertion de messages dans les files d'attente DATA, STATE, EVENT et REPLY appartenant à l'agent

Seuls les agents Managed File Transfer doivent pouvoir utiliser ces files d'attente système. Par conséquent, accordez au groupe FTEAGENT l'accès put, get et inquire. Les noms de ces files d'attente système sont les suivants:

- DONNEES- SYSTEM.FTE.DATA.*nom_agent*
- STATE- SYSTEM.FTE.STATE.*nom_agent*
- EVENT- SYSTEM.FTE.EVENT.*nom_agent*
- REPLY- SYSTEM.FTE.REPLY.*nom_agent*

Par exemple, pour SYSTEM.FTE.DATA.*nom_agent* , utilisez une commande similaire à la suivante:

ULW**Pour les systèmes UNIX, Linux et Windows :**

```
setmqaut -m QM1 -n SYSTEM.FTE.DATA.agent_name -t queue -g FTEAGENT +put +get +inq
```

IBM i**Pour IBM i :**

```
GRTMQMAUT OBJ('SYSTEM.FTE.DATA.agent_name') OBJTYPE(*Q) USER(FTEAGENT) AUT(*PUT)
MQMNAME('QM1')
GRTMQMAUT OBJ('SYSTEM.FTE.DATA.agent_name') OBJTYPE(*Q) USER(FTEAGENT) AUT(*GET)
MQMNAME('QM1')
```

z/OS**Pour z/OS :**

```
RDEFINE MQQUEUE QM1.SYSTEM.FTE.DATA.agent_name UACC(NONE)
PERMIT QM1.SYSTEM.FTE.DATA.agent_name CLASS(MQQUEUE) ID(FTEAGENT) ACCESS(UPDATE)
```

Les agents doivent pouvoir insérer des messages dans les files d'attente de données et de réponses des autres agents. Si des agents sont connectés à des gestionnaires de files d'attente éloignées, vous devrez peut-être accorder des autorisations supplémentaires pour permettre au canal d'insérer des messages dans ces files d'attente.

Droits d'accès sous lesquels le processus d'agent s'exécute

Les droits d'accès sous lesquels le processus d'agent s'exécute affectent les fichiers que l'agent peut lire et écrire à partir du système de fichiers, ainsi que les files d'attente et les rubriques auxquelles l'agent peut accéder. La façon dont les droits sont configurés dépend du système. Ajoutez l'ID utilisateur sous lequel le processus d'agent s'exécute au groupe FTEAGENT. Pour plus d'informations sur l'ajout d'un ID utilisateur à un groupe, voir [Configuration de la sécurité](#) et accédez aux informations de votre système d'exploitation.

Droits d'accès sous lesquels les commandes et IBM MQ Explorer s'exécutent

Les commandes d'administration, par exemple la commande **fteStartAgent**, et le plug-in Managed File Transfer pour IBM MQ Explorer doivent pouvoir insérer des messages dans SYSTEM.FTE.COMMAND.*nom_agent* et extraire les informations publiées de cette file d'attente. Ajoutez les ID utilisateur autorisés à exécuter les commandes ou le IBM MQ Explorer au groupe FTEUSER. Cet ID utilisateur d'émetteur est enregistré dans le journal de transfert. Pour plus d'informations sur l'ajout d'un ID utilisateur à un groupe, voir [Configuration de la sécurité](#) et accédez aux informations de votre système d'exploitation.

Droit d'insertion de messages dans SYSTEM.FTE, file d'attente et SYSTEM.FTE

Seul le processus d'agent doit pouvoir placer des messages sur SYSTEM.FTE, file d'attente et SYSTEM.FTE. Accordez le droit d'insertion, d'obtention et d'interrogation au groupe FTEAGENT sur SYSTEM.FTE, accordez des droits de publication et d'abonnement au groupe FTEAGENT dans SYSTEM.FTE. Exemple :

ULW

Pour les systèmes UNIX, Linux et Windows :

```
setmqaut -m QM1 -n SYSTEM.FTE -t queue -g FTEAGENT +put +get +inq
setmqaut -m QM1 -n SYSTEM.FTE -t topic -g FTEAGENT +pub +sub +resume
```

IBM i

Pour IBM i :

```
GRTMQMAUT OBJ('SYSTEM.FTE') OBJTYPE(*Q) USER(FTEAGENT) AUT(*PUT) MQMNAME('QM1')
GRTMQMAUT OBJ('SYSTEM.FTE') OBJTYPE(*Q) USER(FTEAGENT) AUT(*GET) MQMNAME('QM1')
GRTMQMAUT OBJ('SYSTEM.FTE') OBJTYPE(*TOPIC) USER(FTEAGENT) AUT(*PUB) MQMNAME('QM1')
GRTMQMAUT OBJ('SYSTEM.FTE') OBJTYPE(*TOPIC) USER(FTEAGENT) AUT(*SUB) MQMNAME('QM1')
```

z/OS

Pour z/OS :

```
RDEFINE MQQUEUE QM1.SYSTEM.FTE UACC(NONE)
PERMIT QM1.SYSTEM.FTE CLASS(MQQUEUE) ID(FTEAGENT) ACCESS(UPDATE)
RDEFINE MXTOPIC QM1.PUBLISH.SYSTEM.FTE UACC(NONE)
PERMIT QM1.PUBLISH.SYSTEM.FTE CLASS(MXTOPIC) ID(FTEAGENT) ACCESS(UPDATE)
```

Si des agents sont connectés à des gestionnaires de files d'attente éloignées, des autorisations supplémentaires peuvent également être accordées pour permettre au canal d'insérer des messages dans SYSTEM.FTE, file d'attente.

Pour qu'un message soit publié dans SYSTEM.FTE, enregistrements de droits d'accès de SYSTEM.FTE doit autoriser la publication par l'ID utilisateur contenu dans la structure de descripteur de message (MQMD) du message. Cette procédure est décrite dans la rubrique [Droit de publier des messages de journal et de statut](#).

Permet à un utilisateur de publier des données dans SYSTEM.FTE sous z/OS, vous devez accorder à l'ID utilisateur de l'initiateur de canal l'accès à la publication dans SYSTEM.FTE. Si le profil de sécurité RESLEVEL entraîne la vérification de deux ID utilisateur pour la connexion de l'initiateur de canal, vous devez également accorder l'accès à l'ID utilisateur contenu dans la structure de descripteur de message (MQMD) du message. Pour plus d'informations, voir [Le profil de sécurité RESLEVEL](#)

Droit de recevoir des publications sur SYSTEM.FTE

Les messages de journal de transfert, les messages de progression et les messages d'état sont destinés à une utilisation générale. Par conséquent, accordez au groupe FTEUSER le droit de s'abonner à SYSTEM.FTE . Exemple :

ULW

Pour les systèmes UNIX, Linux et Windows :

```
setmqaut -m QM1 -n SYSTEM.FTE -t topic -g FTEUSER +sub
```

IBM i

Pour IBM i :

```
GRTMQMAUT OBJ('SYSTEM.FTE') OBJTYPE(*TOPIC) USER(FTEUSER) AUT(*SUB) MQMNAME('QM1')
```

z/OS

Pour z/OS :

```
RDEFINE MXTOPIC QM1.SUBSCRIBE.SYSTEM.FTE UACC(NONE)  
PERMIT QM1.SUBSCRIBE.SYSTEM.FTE CLASS(MXTOPIC) ID(FTEUSER) ACCESS(ALTER)
```

Droit de connexion à des gestionnaires de files d'attente éloignées à l'aide de files d'attente de transmission

Dans une topologie de plusieurs gestionnaires de files d'attente, l'agent requiert des droits d'insertion sur les files d'attente de transmission utilisées pour se connecter aux gestionnaires de files d'attente éloignées.

Droit de création d'une file d'attente de réponses temporaire pour les transferts de fichiers

Les demandes de transfert de fichiers attendent la fin du transfert et s'appuient sur la création et le remplissage d'une file d'attente de réponses temporaire. Accordez au groupe FTEUSER les droits DISPLAY, PUT, GET et BROWSE sur la définition de file d'attente modèle temporaire. Exemple :

ULW

Pour les systèmes UNIX, Linux et Windows :

```
setmqaut -m QM1 -n SYSTEM.DEFAULT.MODEL.QUEUE -t queue -g FTEUSER +dsp +put +get +browse
```

IBM i

Pour IBM i :

```
GRTMQMAUT OBJ('SYSTEM.DEFAULT.MODEL.QUEUE') OBJTYPE(*Q) USER(FTEUSER) AUT(*ADM DSP)  
MQMNAME('QM1')  
GRTMQMAUT OBJ('SYSTEM.DEFAULT.MODEL.QUEUE') OBJTYPE(*Q) USER(FTEUSER) AUT(*PUT)  
MQMNAME('QM1')  
GRTMQMAUT OBJ('SYSTEM.DEFAULT.MODEL.QUEUE') OBJTYPE(*Q) USER(FTEUSER) AUT(*GET)  
MQMNAME('QM1')  
GRTMQMAUT OBJ('SYSTEM.DEFAULT.MODEL.QUEUE') OBJTYPE(*Q) USER(FTEUSER) AUT(*BROWSE)  
MQMNAME('QM1')
```

z/OS

Pour z/OS :

```
RDEFINE MQQUEUE QM1.SYSTEM.DEFAULT.MODEL.QUEUE UACC(NONE)  
PERMIT QM1.SYSTEM.DEFAULT.MODEL.QUEUE CLASS(MQQUEUE) ID(FTEUSER) ACCESS(UPDATE)
```

Par défaut, cette file d'attente est SYSTEM.DEFAULT.MODEL.QUEUE, mais vous pouvez configurer le nom en définissant des valeurs pour les propriétés 'modelQueueName' et 'dynamicQueuePrefix' dans le fichier `command.properties`.

Sous z/OS, vous devez également accorder des droits d'accès aux files d'attente temporaires à FTEUSER. Exemple :

```
RDEFINE MQQUEUE QM1.WMQFTE.** UACC(NONE)
PERMIT QM1.WMQFTE.** CLASS(MQQUEUE) ID(FTEUSER) ACCESS(UPDATE)
```

Par défaut, le nom de chaque file d'attente temporaire sous z/OS commence par WMQFTE.

Le tableau suivant récapitule la configuration du contrôle d'accès pour FTEUSER et FTEAGENT dans le schéma de sécurité décrit:

Tableau 30. Récapitulatif de la configuration du contrôle d'accès pour FTEUSER et FTEAGENT

Objet	Type d'objet	UTILISATEURFET	FTEAgent
Gestionnaire de files d'attente d'agent	Gestionnaire de files d'attente		CONNECT, INQ et SETID. ALT_USER est également requis pour activer la vérification des droits utilisateur .
Gestionnaire de files d'attente de coordination	Gestionnaire de files d'attente		
Gestionnaire de files d'attente de commandes	Gestionnaire de files d'attente	Se connecter	Se connecter
SYSTEM.FTE	File d'attente locale		GET et PUT
SYSTEME SYSTEM.FTE.COMMAND. <i>nom_agent</i>	File d'attente locale	PUT	BROWSE, GET, PUT et SETID
SYSTEME SYSTEM.FTE.DATA. <i>nom_agent</i>	File d'attente locale		GET et PUT
SYSTEME SYSTEM.FTE.EVENT. <i>nom_agent</i>	File d'attente locale		BROWSE, GET et PUT
SYSTEME SYSTEM.FTE.REPLY. <i>nom_agent</i>	File d'attente locale		GET et PUT
SYSTEME SYSTEM.FTE.STATE. <i>nom_agent</i>	File d'attente locale		BROWSE, GET, INQ et PUT
SYSTEM.FTE	Rubrique locale	SUBSCRIBE	Abonnement et publication
SYSTEM.DEFAULT.MODEL.QUEUE (ou la file d'attente modèle définie dans Managed File Transfer qui est utilisée pour créer une file d'attente de réponses temporaire.)	File d'attente modèle	BROWSE, DISPLAY, GET et PUT	BROWSE, DISPLAY, GET et PUT
Files d'attente de transmission pour communiquer avec les gestionnaires de files d'attente éloignées	File d'attente locale		PUT

Droits de gestion des transferts via IBM MQ Explorer

Outre l'octroi de droits MFT à des utilisateurs dans des situations déjà mentionnées sur cette page, des droits supplémentaires doivent être accordés à l'utilisateur de l'agent MFT qui administre et exécute toutes les opérations MFT via IBM MQ Explorer. Pour exécuter des commandes telles que la création, l'annulation, la planification du transfert de fichiers, la création, la suppression de moniteurs de ressources et la création de modèles de transfert, l'utilisateur IBM MQ Explorer doit disposer des droits suivants:

- Gestionnaire de files d'attente de coordination: connexion, interrogation, affichage

- Gestionnaire de files d'attente de commandes: connexion, interrogation, affichage
- SYSTEME SYSTEM.FTE : publication, abonnement
- SYSTEME SYSTEM.MQEXPLORER.REPLY.MODEL: display, inquire, get, browse, put
- SYSTEME SYSTEM.ADMIN.COMMAND.QUEUE: inquire, put, display
- SYSTEME SYSTEM.DEFAULT.MODEL.QUEUE: get, put, inquire, display, browse

Pour plus d'informations sur les commandes qui se connectent directement à un gestionnaire de files d'attente, voir [«Les commandes et processus MFT qui se connectent à quel gestionnaire de files d'attente»](#), à la page 376

Référence associée

«Restriction des droits utilisateur sur les actions de l'agent MFT», à la page 368

En plus d'utiliser des groupes pour gérer l'accès aux ressources, vous pouvez activer un niveau supplémentaire de sécurité afin de restreindre les actions d'agent Managed File Transfer qu'un utilisateur peut effectuer. Accordez des autorisations à un utilisateur dans une file d'attente d'autorisations d'agent pour permettre à l'utilisateur d'effectuer des actions d'agent spécifiques.

«Droits d'accès au consignateur MFT», à la page 372

L'utilisateur du système d'exploitation qui exécute le consignateur requiert certains droits IBM MQ sur les files d'attente du consignateur et sur SYSTEM.FTE .

Restriction des droits utilisateur sur les actions de l'agent MFT

En plus d'utiliser des groupes pour gérer l'accès aux ressources, vous pouvez activer un niveau supplémentaire de sécurité afin de restreindre les actions d'agent Managed File Transfer qu'un utilisateur peut effectuer. Accordez des autorisations à un utilisateur dans une file d'attente d'autorisations d'agent pour permettre à l'utilisateur d'effectuer des actions d'agent spécifiques.

Activation de la gestion des autorisations utilisateur

Afin d'activer le contrôle des autorisations utilisateur pour les actions d'agent, procédez comme suit :

1. Dans le fichier `agent.properties`, définissez la valeur `true` pour `authorityChecking`. Pour plus d'informations, voir [«Le fichier MFT agent.properties»](#), à la page 564.
2. Assurez-vous que l'utilisateur qui exécute l'agent possède l'autorisation d'utilisateur alternatif (ALT_USER) IBM MQ pour le gestionnaire de file d'attente d'agent.



Sur la plateforme z/OS, l'utilisateur qui exécute l'agent doit posséder l'autorisation ALT_USER pour les ID utilisateur pouvant demander le droit d'effectuer une action d'agent.

Le même niveau de sécurité doit être activé pour les deux agents impliqués dans un transfert ; en d'autres termes, `authorityChecking` doit avoir la même valeur dans les fichiers de propriétés des deux agents. Les transferts entre les agents dont les valeurs sont différentes pour la propriété `authorityChecking` échoueront.

Files d'attente d'autorisations d'agent

L'agent dispose de files d'attente d'autorisations qui sont utilisées pour contrôler les actions d'agent que les utilisateurs peuvent effectuer. L'agent ne place ni n'obtient aucun message dans ces files d'attente. Les files d'attente d'autorisations d'agent sont les suivantes :

- 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*

Lorsque la gestion des autorisations utilisateur est activée via la définition de la propriété d'agent **authorityChecking=true**, les autorisations auxquelles un utilisateur est associé dans les files d'attente d'autorisations d'agent spécifient les actions qu'il peut effectuer.

Important : V 9.0.0.3 V 9.0.4 Depuis IBM MQ 9.0.0 Fix Pack 3 et IBM MQ 9.0.4, inquire est un droit requis sur toutes les files d'attente de droits d'accès de l'agent.

Le tableau suivant récapitule les droits d'accès IBM MQ dont les utilisateurs ou les groupes ont besoin en plus du droit d'interrogation sur une file d'attente de droits d'accès d'agent pour effectuer des actions spécifiques.

<i>Tableau 31. Niveau d'autorisations d'accès IBM MQ auquel un utilisateur ou un groupe doit être associé dans une file d'attente d'autorisations d'agent pour pouvoir effectuer des actions spécifiques</i>				
Action utilisateur	Droits d'accès Managed File Transfer	Files d'attente d'autorisations	Droits d'accès IBM MQ (Multiplatforms)	RACF Niveau d'accès (z/OS uniquement)
Arrêter l'agent avec l'option -m dans la commande fteStopAgent	Administration	SYSTEM.FTE.AUTHADM1. <i>nom_agent</i>	BROWSE	READ
Démarrer un transfert de fichiers depuis cet agent	Transfer source	SYSTEM.FTE.AUTHTRN1. <i>nom_agent_source</i>	BROWSE	READ
Exécuter un appel géré sur cet agent				
Annuler un transfert de fichier depuis cet agent, démarré par le même utilisateur				
Démarrer un transfert de fichiers vers cet agent	Transfer destination	SYSTEM.FTE.AUTHTRN1. <i>nom_agent_destination</i>	PUT	UPDATE
Annuler un transfert de fichier vers cet agent, démarré par le même utilisateur				
Créer un moniteur de ressources	Surveillance	SYSTEM.FTE.AUTHMON1. <i>nom_agent_moniteur</i>	BROWSE	READ
Supprimer un moniteur de ressources créé par le même utilisateur				
Supprimer un moniteur de ressources créé par un utilisateur quelconque	Monitor operations	SYSTEM.FTE.AUTHOPS1. <i>nom_agent</i>	SET	ALTER
Créer un transfert planifié	Planning	SYSTEM.FTE.AUTHSCH1. <i>nom_agent_source</i>	BROWSE	READ
Supprimer un transfert planifié créé par le même utilisateur				
Supprimer un transfert planifié créé par un utilisateur ou un groupe	Schedule operations	SYSTEM.FTE.AUTHOPS1. <i>nom_agent</i>	PUT	UPDATE

Tableau 31. Niveau d'autorisations d'accès IBM MQ auquel un utilisateur ou un groupe doit être associé dans une file d'attente d'autorisations d'agent pour pouvoir effectuer des actions spécifiques (suite)

Action utilisateur	Droits d'accès Managed File Transfer	Files d'attente d'autorisations	Droits d'accès IBM MQ (Multiplatforms)	RACF Niveau d'accès (z/OS uniquement)
Annuler un transfert créé par l'utilisateur ou le groupe à l'origine du transfert, ou par un autre utilisateur ou groupe	Transfer operations	SYSTEM.FTE.AUTHOPS1.nom_agent_source SYSTEM.FTE.AUTHOPS1.nom_agent_destination	BROWSE	READ

Remarque : Pour accorder à un utilisateur ou à un groupe le droit de configurer un moniteur de ressources ou un transfert planifié qui démarre un transfert, l'utilisateur a besoin des droits `Moniteur` ou `Planification` et des droits source `Transfert` et de destination `Transfert`.

V 9.0.0.3 - V 9.0.4 Depuis IBM MQ 9.0.0 Fix Pack 3 et IBM MQ 9.0.4, l'ordre dans lequel les vérifications des droits d'accès sont effectuées lorsqu'un agent reçoit une demande d'annulation d'un transfert de fichier est modifié de sorte que l'agent vérifie d'abord si l'utilisateur qui demande l'annulation est le même que celui qui a lancé le transfert. Si tel n'est pas le cas, l'agent vérifie alors si l'utilisateur à l'origine de la demande d'annulation possède le droit `browse` dans les files d'attente `SYSTEM.FTE.AUTHOPS1.nom_agent`. La modification de l'ordre des vérifications évite toute erreur inattendue dans les journaux des erreurs de l'agent et du gestionnaire de files d'attente lorsque l'utilisateur qui a demandé le transfert de fichier et l'utilisateur qui a demandé l'annulation est le même.

Un utilisateur peut démarrer un agent et souhaiter qu'il interagisse avec un autre agent. La façon dont les deux agents interagissent dépend du niveau d'autorisation d'accès accordé à l'utilisateur dans l'autre file d'attente d'autorisations d'agent.

Tableau 32. Niveau d'autorisation d'accès IBM MQ auquel l'utilisateur qui démarre un agent doit être associé dans une autre file d'attente d'autorisations d'agent pour que les fichiers puissent être transférés entre les agents

Action d'agent	Droits d'accès Managed File Transfer	Files d'attente d'autorisations	Droits d'accès IBM MQ (Multiplatforms)	RACF Niveau d'accès (z/OS uniquement)
Recevoir un transfert depuis agent_source	Agent source	SYSTEM.FTE.AUTHAGT1.nom_agent_source	BROWSE	READ
Envoyer un transfert à agent_destination	Agent destination	SYSTEM.FTE.AUTHAGT1.nom_agent_destination	PUT	UPDATE

Configuration de la gestion des autorisations utilisateur

Pour autoriser un utilisateur à effectuer une action sur un agent, accordez-lui l'autorisation appropriée dans la file d'attente d'autorisations pertinente. Pour accorder des autorisations à un utilisateur, procédez comme suit :

1. Créez un utilisateur sur le système à l'emplacement du gestionnaire de file d'attente d'agent, en lui attribuant le même nom que l'utilisateur auquel accorder l'autorisation d'effectuer des actions d'agent. Il n'est pas nécessaire que cet utilisateur soit actif.
2. Accordez à l'utilisateur l'autorisation appropriée dans la file d'attente d'autorisations pertinente. Si vous utilisez Linux, UNIX ou Windows, vous pouvez utiliser la commande `setmqaut`.
3. Actualisez la configuration de la sécurité du gestionnaire de files d'attente. Vous pouvez utiliser la commande `MSQC REFRESH SECURITY`.

Exemple

z/OS La commande **setmqaut** n'est pas utilisée sur les systèmes z/OS. Sous z/OS, utilisez RACF à la place. Voir [Configuration de la sécurité sous z/OS](#) pour plus d'informations.

IBM i La commande **setmqaut** n'est pas utilisée sur les systèmes IBM i. Pour IBM i, voir la rubrique [Autorisations d'accès aux objets IBM MQ](#), qui explique comment l'autorisation aux objets IBM MQ est accordée. Trois commandes CL sont disponibles sous IBM i: **Grant MQ Object Authority (GRMQMAUT)**, **Revoke MQ Object Authority (RVKMQMAUT)** et **Refresh MQ Authority (RFRMQMAUT)**.

Un utilisateur membre du groupe `requestor_group` souhaite configurer sur AGENT1 un moniteur de ressources qui transfère un fichier depuis AGENT1, qui s'exécute sous l'utilisateur `user1`, membre du groupe `user1_group`, vers AGENT2, qui s'exécute sous l'utilisateur `user2`, membre du groupe `user2_group`. AGENT1 se connecte à QM1 ; AGENT2 se connecte à QM2. Le contrôle des autorisations est activé sur les deux agents. Pour que ce scénario soit possible, procédez comme suit :

1. `requestor` doit avoir l'autorisation **Monitor** sur AGENT1. Définissez cette autorisation en exécutant la commande suivante sur le système sur lequel QM1 s'exécute :

```
setmqaut -m QM1 -t queue -n SYSTEM.FTE.AUTHMON1.AGENT1 -g requestor_group +browse
```

2. `requestor` doit avoir l'autorisation **Transfer source** sur AGENT1. Définissez cette autorisation en exécutant la commande suivante sur le système sur lequel QM1 s'exécute :

```
setmqaut -m QM1 -t queue -n SYSTEM.FTE.AUTHTRN1.AGENT1 -g requestor_group +browse
```

3. `requestor` doit avoir l'autorisation **Transfer destination** sur AGENT2. Définissez cette autorisation en exécutant la commande suivante sur le système sur lequel QM2 s'exécute :

```
setmqaut -m QM2 -t queue -n SYSTEM.FTE.AUTHTRN1.AGENT2 -g requestor_group +put
```

4. `user2` doit avoir l'autorisation **Agent source** sur AGENT1. Définissez cette autorisation en exécutant la commande suivante sur le système sur lequel QM1 s'exécute :

```
setmqaut -m QM1 -t queue -n SYSTEM.FTE.AUTHAGT1.AGENT1 -g user2_group +browse
```

5. `user1` doit avoir l'autorisation **Agent destination** sur AGENT2. Définissez cette autorisation en exécutant la commande suivante sur le système sur lequel QM2 s'exécute :

```
setmqaut -m QM2 -t queue -n SYSTEM.FTE.AUTHAGT1.AGENT2 -g user1_group +put
```

Journalisation

Si le contrôle des autorisations utilisateur est activé et qu'il échoue, un message de journal indiquant que l'utilisateur n'est pas autorisé est publié dans le gestionnaire de file d'attente de coordination. Pour plus d'informations, voir «[Formats de message MFT pour la sécurité](#)», à la page 834.

Les messages relatifs aux droits utilisateur peuvent être écrits dans le journal des événements de l'agent. Vous pouvez configurer la quantité d'informations écrites dans le journal des événements de l'agent en définissant la propriété `logAuthorityChecks` dans le fichier de propriétés de l'agent. Par défaut, le niveau de contrôle des autorisations est `None`. Vous pouvez aussi définir pour `logAuthorityChecks` la valeur `Failures`, qui spécifie que seuls les échecs du contrôles des autorisations sont rapportés ou `All`, qui spécifie que les réussites ainsi que les échecs des contrôles des autorisations sont rapportés.

Pour plus d'informations, voir «[Le fichier MFT agent.properties](#)», à la page 564.

Référence associée

«[Restriction des droits de groupe pour les ressources spécifiques à MFT](#)», à la page 362

Au lieu d'accorder des droits à des utilisateurs individuels pour tous les différents objets pouvant être impliqués, configurez deux groupes de sécurité pour l'administration du contrôle d'accès Managed File Transfer : FTEUSER et FTEAGENT. Il incombe à l'administrateur IBM MQ de créer et de remplir ces groupes. L'administrateur peut choisir d'étendre ou de modifier la configuration proposée décrite ici.

«Droits d'accès au consignateur MFT», à la page 372

L'utilisateur du système d'exploitation qui exécute le consignateur requiert certains droits IBM MQ sur les files d'attente du consignateur et sur SYSTEM.FTE .

«fteStopAgent», à la page 544

Utilisez la commande **fteStopAgent** pour arrêter un agent Managed File Transfer de manière contrôlée ou pour arrêter un agent immédiatement si nécessaire à l'aide du paramètre **-i** .

Droits d'accès au consignateur MFT

L'utilisateur du système d'exploitation qui exécute le consignateur requiert certains droits IBM MQ sur les files d'attente du consignateur et sur SYSTEM.FTE .

L'utilisateur du système d'exploitation qui exécute le consignateur requiert les droits IBM MQ suivants:

- CONNECT et INQUIRE sur le gestionnaire de files d'attente de coordination.
- Droit SUBSCRIBE sur le système SYSTEM.FTE .
- Droit PUT sur le système SYSTEM.FTE.LOG.RJCT.*nom_journal* file d'attente.
- Droit GET sur SYSTEM.FTE.LOG.CMD.*nom_journal* file d'attente.

Référence associée

«Restriction des droits de groupe pour les ressources spécifiques à MFT», à la page 362

Au lieu d'accorder des droits à des utilisateurs individuels pour tous les différents objets pouvant être impliqués, configurez deux groupes de sécurité pour l'administration du contrôle d'accès Managed File Transfer : FTEUSER et FTEAGENT. Il incombe à l'administrateur IBM MQ de créer et de remplir ces groupes. L'administrateur peut choisir d'étendre ou de modifier la configuration proposée décrite ici.

«Restriction des droits utilisateur sur les actions de l'agent MFT», à la page 368

En plus d'utiliser des groupes pour gérer l'accès aux ressources, vous pouvez activer un niveau supplémentaire de sécurité afin de restreindre les actions d'agent Managed File Transfer qu'un utilisateur peut effectuer. Accordez des autorisations à un utilisateur dans une file d'attente d'autorisations d'agent pour permettre à l'utilisateur d'effectuer des actions d'agent spécifiques.

Droits MFT d'accès aux informations de configuration sensibles

Tout fichier utilisé pour stocker des informations de configuration sensibles, c'est-à-dire tout fichier référencé à partir de l'arborescence de configuration IBM MQ , ne doit pas disposer de droits de lecture, d'écriture ou (le cas échéant) de suppression à l'échelle du système. Ces restrictions s'appliquent également aux fichiers de clés certifiées et aux fichiers de clés.

Si un processus Managed File Transfer détecte une condition selon laquelle un fichier de configuration contient des informations sensibles, est un fichier de clés ou de clés certifiées et dispose de droits de lecture, d'écriture ou de suppression à l'échelle du système, le processus effectue l'une des actions suivantes:

- Echec du démarrage, si la condition est détectée au moment du démarrage.
- Génère un message d'avertissement et ignore le contenu du fichier de configuration si la condition a été détectée lors de l'exécution. Cela concerne le pont de protocole et le pont Connect:Direct , qui rechargent une configuration si elle est modifiée pendant que le processus est en cours d'exécution.

Sur les systèmes dotés d'un système de fichiers de type UNIX

Les critères permettant de déterminer qu'un fichier possède des droits d'accès inacceptables à l'échelle du système sont les suivants:

- Le droit de lecture a été accordé à la classe others sur le fichier
- Le droit d'accès en écriture a été accordé à la classe others sur le fichier

- Le droit d'accès en écriture a été accordé à la classe others sur le répertoire contenant le fichier

Sur les systèmes Windows

Les critères permettant de déterminer qu'un fichier possède des droits d'accès inacceptables à l'échelle du système sont les suivants:

- L'un des groupes Tout le monde, Invités ou Utilisateurs possède l'un des droits suivants:
 - Droit de lecture des données sur le fichier
 - Ajouter le droit d'accès aux données sur le fichier
 - Droit d'accès en écriture aux données sur le fichier
- L'un des groupes Tous, Invités ou Utilisateurs dispose des droits de création de fichiers sur le dossier contenant le fichier, ainsi que des droits suivants:
 - Droits de suppression de sous-dossiers et de fichiers sur le dossier contenant le fichier
 - Droit de suppression sur le fichier

Droits de publication du journal et des messages d'état des agents MFT

Les agents Managed File Transfer émettent divers messages de journal, de progression et de statut qui sont publiés sur le gestionnaire de files d'attente de coordination. La publication de ces messages est soumise au modèle de sécurité IBM MQ et, dans certains cas, vous devrez peut-être effectuer une configuration supplémentaire pour activer la publication.

Pour plus d'informations sur la sécurité IBM MQ , voir la section commençant par [Sécurisation](#).

Les agents Managed File Transfer envoient des messages à des fins de publication à SYSTEM.FTE FTE sur le gestionnaire de file d'attente de coordination. Chaque message contient un ID utilisateur dans son descripteur de message (MQMD). Les messages sont publiés à l'aide d'un objet de rubrique également appelé SYSTEM.FTE. Pour la publication d'un message donné, les enregistrements de droits d'accès de SYSTEM.FTE doit permettre la publication par l'ID utilisateur contenu dans le MQMD du message.

Sous z/OS, l'ID utilisateur de l'initiateur de canal doit avoir accès à la publication dans SYSTEM.FTE . L'ID utilisateur dans le MQMD du message doit également avoir accès à la publication dans cette rubrique si le [profil de sécurité RESLEVEL](#) entraîne la vérification de deux ID utilisateur pour la connexion de l'initiateur de canal.

L'ID utilisateur initialement contenu dans le message dépend de la manière dont l'agent est connecté à son propre gestionnaire de files d'attente. Les messages provenant des agents connectés aux liaisons contiennent l'ID utilisateur sous lequel l'agent s'exécute. Les messages des agents connectés au client contiennent un ID utilisateur IBM MQ interne.

Vous pouvez modifier l'ID utilisateur dans un message. Pour les agents connectés au client et aux liaisons, vous pouvez utiliser la propriété publicationMDUser (dans le fichier agent . properties) pour spécifier un ID utilisateur, qui est utilisé dans tous les messages de journal et de statut de cet agent. L'agent doit être autorisé par son propre gestionnaire de files d'attente à utiliser cet ID utilisateur alternatif ; accordez ce droit en accordant des droits d'accès à l'ID utilisateur sous lequel l'agent s'exécute.

Vous pouvez également modifier l'ID utilisateur contenu dans tous les messages d'un agent connecté au client à l'aide de la propriété MCAUSER sur le canal utilisé par l'agent pour se connecter à son gestionnaire de files d'attente.

Vous pouvez modifier l'ID utilisateur dans les messages à l'aide d'un exit de canal, par exemple sur le canal récepteur qui introduit des messages dans le gestionnaire de files d'attente de coordination.

En fonction de la topologie et des règles IBM MQ , l'administrateur IBM MQ peut utiliser les informations de cette rubrique de différentes manières pour s'assurer que la publication des messages de statut et de journal a lieu. Voici deux exemples:

- Déterminez tous les ID utilisateur utilisés par les agents du réseau. Accordez explicitement un enregistrement de droits d'accès pour chacun de ces ID.
- Créez un ou plusieurs noms d'utilisateur communs pour publier les messages de journal et de statut. Créez des enregistrements de droits d'accès pour ces noms d'utilisateur sur le gestionnaire de files d'attente de coordination. Définissez la propriété publicationMDUser de chaque agent sur un nom d'utilisateur commun. Sur chaque gestionnaire de files d'attente d'agent, accordez le droit setid à l'ID utilisateur sous lequel l'agent s'exécute pour lui permettre d'accepter la propriété publicationMDUser .

Droits d'accès de MFT aux systèmes de fichiers

Pour toute demande de transfert de fichier, les processus de l'agent Managed File Transfer requièrent un certain niveau d'accès à leurs systèmes de fichiers locaux.

- Pour effectuer un transfert à partir d'un fichier source, l'ID utilisateur sous lequel l'agent source s'exécute doit disposer d'un accès en lecture au fichier source. En outre, vous devrez peut-être accorder à l'agent source le droit de suppression ou d'écriture en fonction de l'attribut de disposition de la source.
- Pour effectuer un transfert vers un fichier ou un répertoire, l'ID utilisateur sous lequel l'agent de destination s'exécute doit disposer du droit d'écriture sur le chemin indiqué. En outre, vous devrez peut-être accorder à l'agent de destination le droit de mise à jour, en fonction de l'attribut d'existence de la destination.
- En plus des droits d'accès aux fichiers que vous accordez au processus d'agent, vous pouvez également utiliser le bac à sable pour spécifier et appliquer une zone de chemin de fichier restreint. Pour plus d'informations, voir la section «MFT bacs à sable», à la page 46.
- Si les fichiers que vous souhaitez transférer vers ou depuis ne se trouvent pas dans un emplacement accessible à l'agent, par exemple  un fichier VSAM ou dans un emplacement limité par la fonction de bac à sable, vous pouvez utiliser des exits utilisateur Managed File Transfer pour déplacer le fichier vers ou depuis un emplacement accessible par l'agent. Pour plus d'informations, voir la section «Personnalisation de MFT avec des exits utilisateur», à la page 263.

Propriété commandPath MFT

La propriété commandPath permet de spécifier les emplacements à partir desquels Managed File Transfer peut exécuter des commandes. Soyez extrêmement prudent lorsque vous définissez cette propriété car toute commande de l'un des emplacements commandPaths spécifiés peut être effectivement appelée à partir d'un système client distant qui peut envoyer des commandes à l'agent.

Vous pouvez spécifier une commande à exécuter sur le système sur lequel l'agent est exécuté à partir des fonctions de transfert géré et d'appel géré de Managed File Transfer. Pour plus d'informations, voir [Appel du programme](#). Toutefois, les commandes doivent se trouver sur des chemins référencés par la propriété d'agent commandPath.

Si la commande spécifiée n'est pas complète, Managed File Transfer tente de rechercher une commande correspondante dans le chemin d'accès aux commandes. Si ce dernier contient plusieurs commandes correspondantes, la première occurrence est utilisée.

Par défaut, la propriété commandPath est vide de sorte que l'agent ne puisse pas appeler de commandes.

Spécifiez la propriété d'agent commandPath comme suit :

```
commandPath=command_directory_name
separator...command_directory_name
```

 Ou, pour z/OS uniquement, spécifiez :

```
commandPath=command_directory_name_or_data_set_name_prefix
separator...command_directory_name_or_data_set_name_prefix
```

où :

- *nom_répertoire_commandes* représente un chemin de répertoire pour les commandes qui peuvent être exécutées.
-  *nom_répertoire_commandes_ou_préfixe_nom_fichier* représente un chemin de répertoire z/OS UNIX System Services pour les commandes qui peuvent être exécutées, ou un préfixe de nom de fichier, qui commence par //. Vous pouvez choisir d'utiliser un préfixe de nom de fichier qualifié complet ou non qualifié (c'est-à-dire au format //'HLQ. . .' ou //HLQ. . .). Spécifiez les fichiers partitionnés sous la forme //'HLQ() . . .' ou //HLQ() . . . Utilisez des fichiers pour spécifier uniquement des commandes de script JCL.
- *séparateur* représente le séparateur spécifique à la plateforme.

 Par exemple, sur un système UNIX, si vous souhaitez exécuter des commandes qui se trouvent dans les répertoires /home/user1/cmds1 et /home/user1/cmds2, définissez la propriété d'agent `commandPath` comme suit :

```
commandPath=/home/user1/cmds1:/home/user1/cmds2
```

 Par exemple, sur un système Windows, si vous souhaitez exécuter des commandes qui se trouvent dans les répertoires C:\File Transfer\commands et C:\File Transfer\agent commands, définissez la propriété d'agent `commandPath` comme suit :

```
commandPath=C:\\File Transfer\\commands;C:\\File Transfer\\agent commands
```

Sur un système Windows, le caractère de séparation, barre oblique inversée (\), doit être mis en échappement et entré sous la forme d'une double barre oblique inversée (\\). La barre oblique inversée (\) peut également être remplacée par une barre oblique (/).

 Par exemple, sous z/OS, si vous souhaitez exécuter des commandes qui sont :

- Dans les répertoires /home/user1/cmds1 et /home/user1/cmds2
 - Dans les fichiers commençant par //'USER.CMD1', //CMD2,
 - Des membres d'un ensemble de données partitionnées complet intitulé //'USER.CMDS'
- définissez la propriété d'agent `commandPath` comme suit :

```
commandPath=/home/user1/cmds1:/home/user1/cmds2://'USER.CMD1'://CMD2://'USER.CMDS()'
```

Important : Soyez extrêmement prudent lorsque vous définissez cette propriété car toute commande de l'un des emplacements `commandPaths` spécifiés peut être appelée à partir d'un système client distant qui peut envoyer des commandes à l'agent. Pour cette raison, par défaut, lorsque vous spécifiez un emplacement `commandPath`, l'utilisation des bacs à sable est configuré de sorte que l'accès soit automatiquement refusé à tous les répertoires `commandPath` (et leurs sous-répertoires) pour un transfert :

- Si l'agent est configuré pour utiliser un bac à sable d'agent, les répertoires `commandPath` sont automatiquement ajoutés à la liste des répertoires refusés au démarrage de l'agent.
- Si l'agent est configuré avec un ou plusieurs bacs à sable utilisateur, les répertoires `commandPath` sont ajoutés en tant qu'éléments `<exclude>` aux éléments `<read>` et `<write>` pour chaque bac à sable utilisateur au démarrage de l'agent.
- Si l'agent n'est pas configuré pour utiliser un bac à sable d'agent ou des bacs à sable d'utilisateur, un bac à sable d'agent est créé au démarrage de l'agent et les répertoires `commandPath` sont spécifiés comme répertoires refusés.

 Vous pouvez remplacer ce comportement à des fins de compatibilité avec les éditions suivantes :

- IBM WebSphere MQ File Transfer Edition.

- Le composant IBM WebSphere MQ 7.5.0 Fix Pack 1 Managed File Transfer (ou version antérieure).
- Le composant IBM WebSphere MQ 7.5.0 Fix Pack 2 Managed File Transfer (ou version ultérieure) sur une installation pour laquelle la propriété d'installation `enableFunctionalFixPack=7502` n'est pas définie.

Vous pouvez remplacer ce comportement en ajoutant la propriété suivante au fichier `agent.properties` :

```
addCommandPathToSandbox=false
```

V9.0.0.1 Si la propriété `addCommandPathToSandbox` est présente et définie sur `false`, le comportement suivant est constaté :

- Si l'agent est configuré pour utiliser un bac à sable d'agent, et qu'aucun répertoire autorisé n'est spécifié pour ce bac à sable, les répertoires `commandPath` sont automatiquement ajoutés à la liste des répertoires refusés au démarrage de l'agent.
- Si l'agent est configuré pour utiliser un bac à sable d'agent, et qu'un ou plusieurs répertoires autorisés sont spécifiés pour ce bac à sable, les répertoires `commandPath` ne sont pas ajoutés à la liste des répertoires refusés au démarrage de l'agent.
- Si l'agent est configuré avec un ou plusieurs bacs à sable utilisateur, les bacs à sable utilisateur ne sont pas modifiés et les répertoires `commandPath` ne sont pas ajoutés en tant qu'éléments `<exclude>` aux éléments `<read>` et `<write>` pour chaque bac à sable utilisateur.
- Si l'agent n'est pas configuré pour utiliser un bac à sable d'agent ou des bacs à sable d'utilisateur, un bac à sable d'agent est créé au démarrage de l'agent et les répertoires `commandPath` sont spécifiés comme répertoires refusés.

Référence associée

«Le fichier MFT `agent.properties`», à la page 564

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.

Les commandes et processus MFT qui se connectent à quel gestionnaire de files d'attente

Une topologie Managed File Transfer se compose d'un certain nombre de composants différents.

Ces composants sont les suivants:

- Un ou plusieurs agents, avec leur gestionnaire de files d'attente d'agent associé
- Un gestionnaire de files d'attente de coordination
- Un gestionnaire de files d'attente de commandes
- Un certain nombre de commandes utilisées pour administrer la topologie et soumettre des transferts gérés
- Un consignateur facultatif, qui collecte des informations sur les transferts gérés effectués par les agents dans la topologie
- Le plug-in IBM MQ Explorer Managed File Transfer, qui peut être utilisé pour effectuer certaines tâches d'administration et afficher des informations sur les transferts gérés.

Les agents, les consignateurs, les commandes et le plug-in IBM MQ Explorer Managed File Transfer se connectent à un ou plusieurs gestionnaires de files d'attente lorsqu'ils s'exécutent.

Les tableaux suivants récapitulent les agents de gestionnaire de files d'attente, les consignateurs, les commandes et le plug-in IBM MQ Explorer Managed File Transfer auxquels ils se connectent lorsqu'ils s'exécutent.

S'il n'y a pas de caractères X pour une commande ou un processus dans la table, la commande ne se connecte à aucun gestionnaire de files d'attente ou processus lorsqu'elle est exécutée.

Tableau 33. Récapitulatif des commandes Managed File Transfer qui se connectent à quel gestionnaire de files d'attente

Nom de la commande	Gestionnaire de files d'attente d'agent	Gestionnaire de files d'attente de commandes	Gestionnaire de files d'attente de coordination	Gestionnaire de files d'attente du consignateur
fteAnt				
fteCancelTransfer		X		
fteChangeDefaultConfigurationOptions				
fteCleanAgent	X			
fteCreateAgent	X			
fteCreateBridgeAgent	X			
fteCreateCDAgent	X			
V 9.0.1 fteCreateEnvironnement				
fteCreateLogger				
fteCreateMonitor		X		
fteCreateTemplate			X	
fteCreateTransfer		X		
fteDefine				
fteDelete				
fteDeleteAgent	X		X	
fteDeleteLogger				
fteDeleteMonitor		X		
fteDeleteScheduledTransfer		X		
fteDeleteTemplates			X	
fteDisplayVersion				
fteListAgents			X	
fteListMonitors			X	
fteListScheduledTransfers			X	
fteListTemplates			X	
fteMigrateAgent				
fteMigrateConfigurationOptions				
fteMigrateLogger				
fteModifyAgent				
fteModifyLogger				
fteObfuscate				
ftePingAgent		X		
fteRAS				
V 9.0.0 fteSetAgentLogNiveau				
fteSetAgentTraceLevel				
fteSetLoggerTraceLevel				

Tableau 33. Récapitulatif des commandes Managed File Transfer qui se connectent à quel gestionnaire de files d'attente (suite)

Nom de la commande	Gestionnaire de files d'attente d'agent	Gestionnaire de files d'attente de commandes	Gestionnaire de files d'attente de coordination	Gestionnaire de files d'attente du consignateur
 fteSetProductID				
fteSetupCommands				
fteSetupCoordination				
fteShowAgentDetails			X	
fteShowLoggerDetails				
fteStartAgent				
fteStartLogger				
fteStopAgent		X		
fteStopLogger		X		

Tableau 34. Récapitulatif des processus Managed File Transfer qui se connectent à quel gestionnaire de files d'attente

Processus	Gestionnaire de files d'attente d'agent	Gestionnaire de files d'attente de commandes	Gestionnaire de files d'attente de coordination	Gestionnaire de files d'attente du consignateur
Agents Managed File Transfer	X			
Managed File Transfer plug-in pour IBM MQ Explorer		X	X	
Managed File Transfer Logger			X	X

Le fichier qui contient les données d'identification requises pour la connexion à chaque type de gestionnaire de files d'attente, c'est-à-dire les gestionnaires de files d'attente d'agent, de commande et de coordination, peut être spécifié dans le fichier de propriétés associé. Par exemple, le gestionnaire de files d'attente de coordination possède un fichier `coordination.properties`. Dans ce fichier, vous pouvez définir la propriété `coordinationQMgrAuthenticationCredentialsFile` pour qu'elle pointe vers le fichier de données d'identification.

Les commandes qui se connectent au gestionnaire de file d'attente de coordination utilisent les données d'identification spécifiées dans ce fichier. Si la sécurité est activée sur un gestionnaire de files d'attente et que cette propriété n'est pas correctement définie, les commandes MFT ne s'exécutent pas correctement. Pour plus d'informations, voir [Authentification des connexions MFT et IBM MQ](#).

Information associée

[Jeux de commandes MFT installés](#)

Détails sur les commandes MFT qui se connectent à quel gestionnaire de files d'attente

Informations supplémentaires sur les commandes Managed File Transfer qui se connectent à quel gestionnaire de files d'attente.

Cette rubrique développe les informations de la rubrique [«Les commandes et processus MFT qui se connectent à quel gestionnaire de files d'attente»](#), à la page 376, ainsi que des illustrations.

Commandes qui se connectent au gestionnaire de files d'attente de coordination

Les commandes suivantes se connectent au gestionnaire de file d'attente de coordination:

- [fteCreateTemplate](#)
- [fteDeleteAgent](#)

- [fteDeleteTemplates](#)
- [fteListAgents](#)
- [fteListMonitors](#)
- [fteListScheduledTransfers](#)
- [fteListTemplates](#)
- [fteShowAgentDetails](#)

Le gestionnaire de files d'attente de coordination pour une topologie Managed File Transfer est un concentrateur central qui a une connaissance de l'ensemble de la topologie. Le gestionnaire de files d'attente de coordination est connecté à tous les gestionnaires de files d'attente d'agent d'une topologie via les canaux émetteur et récepteur. Les agents publient régulièrement des informations de statut dans le gestionnaire de files d'attente de coordination et y stockent leurs modèles de transfert.

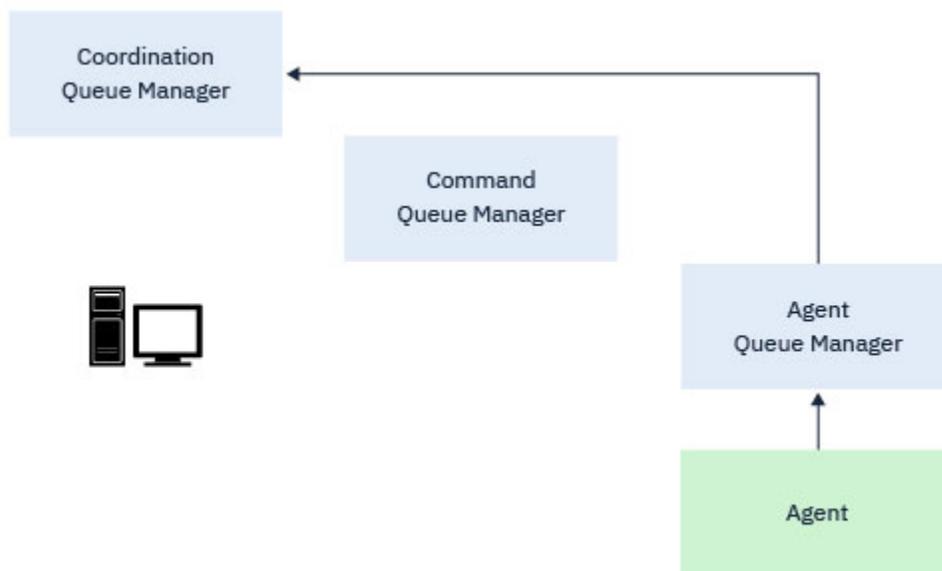


Figure 9. Les agents publient des informations de statut ou stockent des modèles de transfert sur le gestionnaire de files d'attente de coordination

Lorsque l'une des commandes précédentes qui se connectent au gestionnaire de files d'attente de coordination est exécutée, elles se connectent directement au gestionnaire de files d'attente de coordination et:

- Créer ou supprimer un modèle de transfert.
- Interrogation des informations d'état sur les agents, les moniteurs ou les transferts planifiés et affichage de ces informations à l'utilisateur.

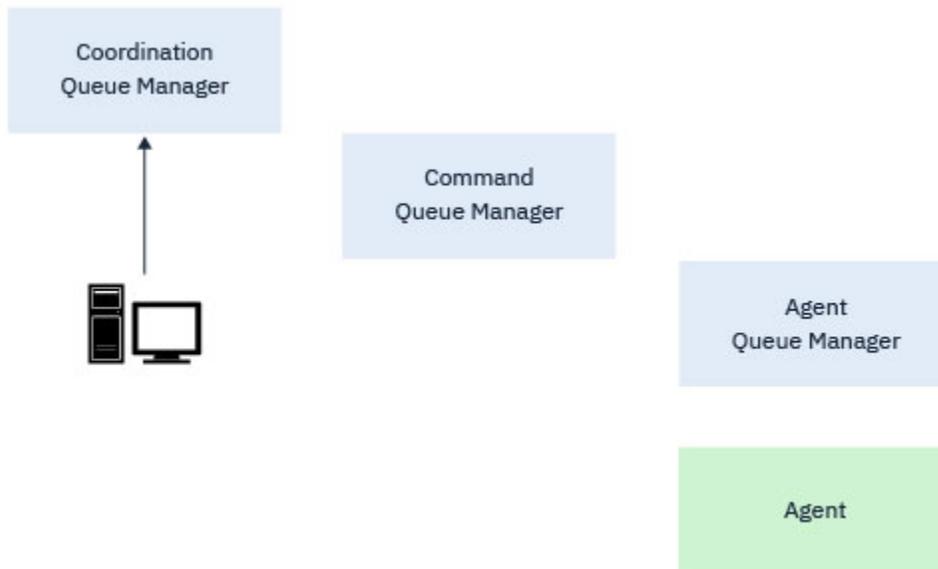


Figure 10. Les commandes se connectent ensuite directement au gestionnaire de file d'attente de coordination pour obtenir les informations de statut appropriées ou utiliser des modèles de transfert.

Commandes qui se connectent au gestionnaire de files d'attente de commandes

Les commandes suivantes se connectent au gestionnaire de files d'attente de commandes:

- [fteCancelTransfer](#)
- [fteCreateMonitor](#)
- [fteCreateTransfer](#)
- [fteDeleteMonitor](#)
- [fteDeleteScheduledTransfer](#)
- [ftePingAgent](#)
- [fteStopAgent](#)

Vous pouvez considérer le gestionnaire de files d'attente de commandes comme une forme de passerelle dans une topologie Managed File Transfer . Il est connecté aux gestionnaires de files d'attente d'agent à l'aide des canaux émetteur et récepteur.

Lorsque l'une des commandes précédentes qui se connectent au gestionnaire de files d'attente est exécutée, elles:

- Connectez-vous au gestionnaire de files d'attente de commandes.
- Créez une file d'attente de réponses temporaire.
- Envoyez un message contenant les détails de la commande à l'agent approprié.

Le message est acheminé via le réseau IBM MQ vers le gestionnaire de files d'attente de l'agent, où il est prélevé par l'agent et traité.

Une fois que l'agent a traité la commande, il renvoie une réponse au gestionnaire de files d'attente de commandes, où la réponse est prise en compte par la commande.

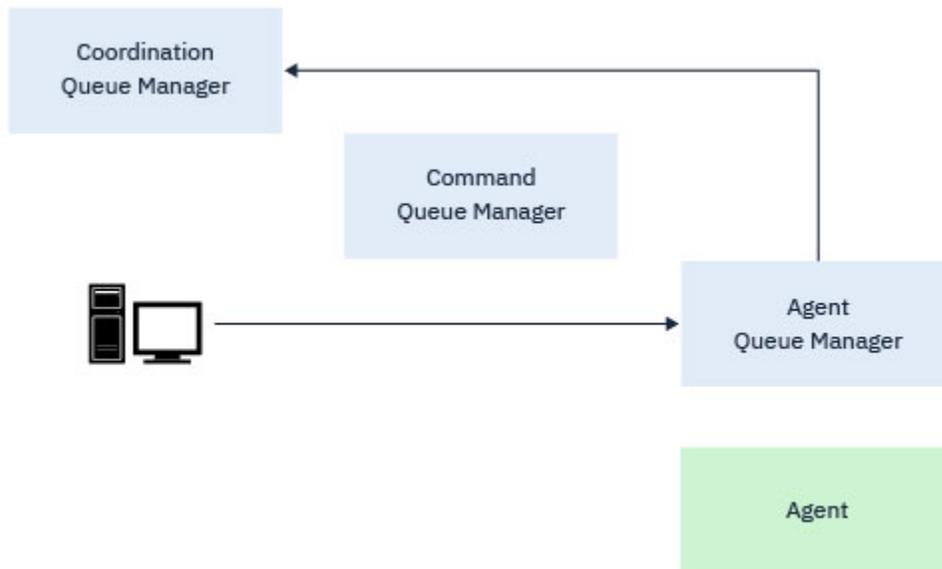


Figure 12. Les différentes commandes **fteCreate**, ainsi que la commande **fteDeleteAgent**, se connectent au gestionnaire de files d'attente d'agent et envoient un message au gestionnaire de files d'attente de coordination pour enregistrer ou désenregistrer un agent de la topologie.



Avertissement : fteCleanAgent se connecte au gestionnaire de files d'attente d'agent et supprime les informations d'état de cet agent de ses files d'attente système.

L'exécution de cette commande peut avoir un impact sur l'ensemble de la topologie. Par conséquent, vous ne devez exécuter cette commande que sur instruction de IBM.

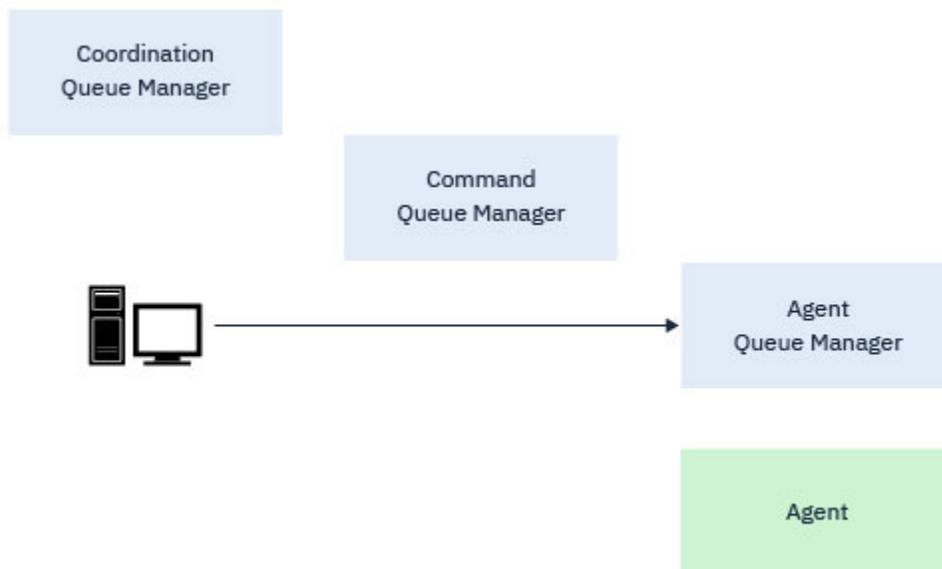


Figure 13. La commande **fteCleanAgent** se connecte directement au gestionnaire de files d'attente d'agent et supprime les informations d'état des files d'attente système de l'agent

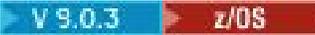
Information associée

[Jeux de commandes MFT installés](#)

Commandes MFT

Toutes les commandes Managed File Transfer sont répertoriées avec des liens vers leurs descriptions détaillées.

Tableau 35. Commandes Managed File Transfer et leur fonction	
Nom de la commande	Objet
Commandes de migration:	
fteMigrateAgent	Migration d'un agent et de sa configuration depuis IBM WebSphere MQ File Transfer Edition 7.0.4 vers Managed File Transfer 7.5 ou version ultérieure
fteMigrateConfigurationOptions	Migrez une configuration IBM WebSphere MQ File Transfer Edition 7.0 vers Managed File Transfer dans IBM WebSphere MQ 7.5 ou version ultérieure.
fteMigrateLogger	Migrez la configuration d'un consignateur de base de données autonome depuis IBM WebSphere MQ File Transfer Edition 7.0.1 ou une version ultérieure vers Managed File Transfer 7.5 ou une version ultérieure.
Commandes de configuration:	
fteChangeDefaultConfigurationOptions	Modification des options de configuration par défaut que vous souhaitez que Managed File Transfer utilise
fteCreateAgent	Création d'un Managed File Transfer Agent
fteCreateBridgeAgent	Création d'un agent de pont de protocole Managed File Transfer
fteCreateCDAgent	Création d'un agent de pont Managed File Transfer Connect:Direct
V 9.0.1 fteCreateEnvironnement	Définissez la variable d'environnement pour la configuration et le transfert des fichiers pour Redistributable Managed File Transfer Agent.
fteCreateLogger	Création d'un consignateur Managed File Transfer
fteDefine	Générez les scripts de configuration nécessaires pour définir les objets spécifiés.
fteDelete	Générez les scripts de configuration nécessaires pour supprimer les objets spécifiés.
fteDeleteAgent	Suppression d'un Managed File Transfer Agent particulier
fteDeleteLogger	Suppression d'un consignateur Managed File Transfer
fteModifyAgent	Windows uniquement. Modifiez un agent, un agent de pont Connect:Direct ou un agent de pont de protocole pour qu'il s'exécute en tant que service Windows .
fteModifyLogger	Windows uniquement. Modifiez le consignateur pour qu'il s'exécute en tant que service Windows .
fteSetupCommands	Indiquez les détails du gestionnaire de files d'attente qui se connecte au réseau IBM MQ lorsque vous émettez des commandes
fteSetupCoordination	Configuration d'un gestionnaire de files d'attente de coordination Managed File Transfer
Commandes d'administration:	
fteAnt	Exécutez un script Ant dans un environnement avec les tâches Ant de transfert de fichier disponibles.
fteCancelTransfer	Annuler un transfert de fichier
fteCleanAgent	Nettoyer les files d'attente utilisées par un agent
fteCreateMonitor	Créer et démarrer un nouveau moniteur de ressources
fteCreateTemplate	Créer un modèle de transfert pour une utilisation ultérieure

Tableau 35. Commandes Managed File Transfer et leur fonction (suite)	
Nom de la commande	Objet
fteCreateTransfer	Créer et démarrer un nouveau transfert de fichiers
fteDeleteMonitor	Arrêter et supprimer un moniteur de ressources existant
fteDeleteScheduledTransfer	Supprimer un transfert de fichier particulier que vous avez précédemment planifié
fteDeleteTemplates	Supprimer les modèles de transfert de fichiers existants
fteListAgents	Liste de tous les agents enregistrés sur un gestionnaire de files d'attente de coordination particulier
fteListMonitors	Liste de tous les moniteurs de ressources enregistrés pour un gestionnaire de files d'attente de coordination particulier
fteListScheduledTransfers	Répertoriez tous les transferts Managed File Transfer que vous avez précédemment créés à l'aide de la ligne de commande ou du IBM MQ Explorer.
fteListTemplates	Répertorier tous les modèles de transfert de fichiers pour un gestionnaire de files d'attente de coordination
ftePingAgent	Envoie une commande PING à un agent pour déterminer si l'agent est actif et s'il peut traiter les transferts.
 fteSetProductID	Définir l'ID de produit d'enregistrement SCRT z/OS
 fteShowAgentDetails	Afficher les détails d'un agent particulier
fteShowLoggerDetails	Afficher les détails d'un consignateur particulier
fteStartAgent	Démarrer un agent particulier avant de l'utiliser pour transférer des fichiers
fteStartLogger	Démarrer le consignateur
fteStopAgent	Arrêter un agent particulier
fteStopLogger	Arrêter le consignateur
Commande de sécurité:	
fteObfuscate	Chiffrez les données sensibles dans les fichiers de données d'identification.
Commandes d'identification et de résolution des problèmes:	
fteDisplayVersion	Afficher la version du produit
 fteSetAgentLogNiveau	Activez ou désactivez la consignation des informations de diagnostic pour les transferts de fichiers entre un agent de pont de protocole Managed File Transfer et les serveurs de fichiers FTP/SFTP/FTPS.
fteSetAgentTraceLevel	Définir le niveau de trace de l'agent à exécuter
fteSetLoggerTraceLevel	Définir le niveau de trace du consignateur à exécuter
fteRAS	Exécuter l'outil de collecte RAS

Voir [Installed MFT command sets](#) pour un tableau indiquant les commandes qui sont installées avec l'offre Managed File Transfer .

La syntaxe de chaque commande et de ses paramètres est présentée sous la forme d'un diagramme de syntaxe appelé diagramme de chemin de fer. Pour plus d'informations sur l'interprétation des diagrammes de chemin de fer, voir [Comment lire les diagrammes de chemin de fer.](#)

Droits d'utilisation des commandes MFT

Votre ID utilisateur doit être membre du groupe mqm si vous souhaitez émettre des commandes Managed File Transfer, sauf si vous avez déjà configuré IBM MQ pour permettre aux utilisateurs qui ne font pas partie du groupe mqm d'émettre des commandes.

 Pour plus d'informations sur la définition d'un groupe alternatif à mqm sur z/OS, voir [Emission de commandes pour IBM MQ for z/OS](#)

Pour plus d'informations sur l'autorisation, voir [Droit d'administration d' IBM MQ](#).  Si vous utilisez IBM i, commencez par la rubrique suivante: [IBM MQ droits d'accès](#).

Un sous-ensemble des commandes Managed File Transfer peut être émis à l'aide de l'explorateur IBM MQ.

Emission de commandes à partir des systèmes Windows et UNIX

Notez les informations suivantes spécifiques à l'environnement pour l'émission de commandes:

Managed File Transfer pour Windows

Toutes les commandes peuvent être émises à partir d'une ligne de commande. Les noms de commande ne sont pas sensibles à la casse: vous pouvez les entrer en majuscules, en minuscules ou en combinaison de majuscules et de minuscules. Toutefois, les arguments des commandes de contrôle (telles que les noms de file d'attente) et les paramètres (tels que **-m** pour le nom du gestionnaire de files d'attente) sont sensibles à la casse.

Dans les descriptions de syntaxe, le trait d'union (-) est utilisé comme indicateur d'indicateur.

Managed File Transfer pour les systèmes UNIX

Toutes les commandes Managed File Transfer peuvent être émises à partir d'un shell. Toutes les commandes sont sensibles à la casse.

Emission de commandes à partir de systèmes z/OS



Les commandes Managed File Transfer sont installées dans le sous-répertoire bin de l'emplacement choisi lors de l'installation du produit. Les commandes peuvent être exécutées à partir de l'une des options suivantes:

- Directement à partir de l'environnement USS en spécifiant le chemin d'accès à la commande ou en incluant le sous-répertoire bin dans le chemin de la commande utilisateur.
- A partir d'un ensemble de données PDSE de commandes configurées à partir de la bibliothèque de modèles de commande PDSE, pour un agent ou un consignateur particulier. Pour plus d'informations, voir [«Création d'un fichier de commandes d'agent ou de consignateur MFT»](#), à la page 64.

Emission de commandes à partir de la plateforme IBM i



Notez les informations suivantes spécifiques à l'environnement pour l'émission de commandes sous IBM i:

- Vous pouvez démarrer les commandes Managed File Transfer à l'aide de l'interpréteur Qshell. Pour démarrer l'interpréteur Qshell, émettez la commande **STRQSH** à partir d'une ligne de commande système IBM i.
- Lorsque vous exécutez des commandes dans l'environnement Qshell, les noms de commande ne sont pas sensibles à la casse: vous pouvez les entrer en majuscules, en minuscules ou en combinaison de majuscules et de minuscules. Toutefois, les arguments des commandes de contrôle (telles que les noms de file d'attente) et les paramètres (tels que **-m** pour le nom du gestionnaire de files d'attente) sont sensibles à la casse.

Référence associée

[«Codes retour pour MFT»](#), à la page 347

Les commandes Managed File Transfer , les tâches Ant et les messages de journal fournissent des codes retour indiquant si les fonctions ont abouti.

Comment lire les diagrammes de chemin de fer (syntaxe)

Chaque diagramme de chemin de fer commence par une double flèche droite et se termine par une paire de flèches droite et gauche. Les lignes commençant par une seule flèche droite sont des lignes de continuation. Vous lisez un diagramme de chemin de fer de gauche à droite et de haut en bas, en suivant la direction des flèches.

Les autres conventions utilisées dans les diagrammes de chemin de fer sont les suivantes:

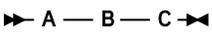
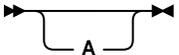
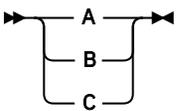
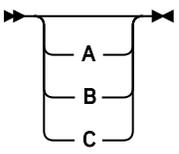
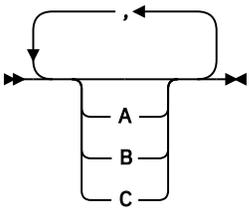
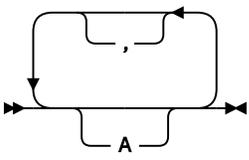
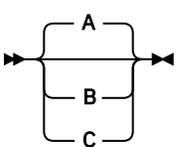
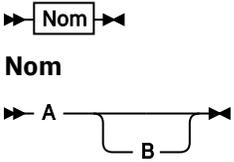
Convention	Explication
	<p>Vous devez spécifier les valeurs A, B et C. Les valeurs requises sont affichées sur la ligne principale d'un diagramme de chemin de fer.</p>
	<p>Vous pouvez spécifier la valeur A. Les valeurs facultatives sont affichées sous la ligne principale d'un diagramme de chemin de fer.</p>
	<p>Les valeurs A, B et C sont des alternatives que vous devez spécifier.</p>
	<p>Les valeurs A, B et C sont des alternatives que vous pouvez spécifier.</p>
	<p>Vous pouvez spécifier une ou plusieurs des valeurs A, B et C. Tout séparateur requis pour les valeurs multiples ou répétées (dans cet exemple, la virgule (,)) est affiché sur la flèche.</p>
	<p>Vous pouvez spécifier la valeur A plusieurs fois. Le séparateur de cet exemple est facultatif.</p>
	<p>Les valeurs A, B et C sont des alternatives que vous pouvez spécifier. Si vous n'indiquez aucune des valeurs affichées, la valeur par défaut A (la valeur affichée au-dessus de la ligne principale) est utilisée.</p>

Tableau 36. Comment lire les diagrammes de chemin de fer (suite)

Convention	Explication
 <p>Nom</p>	Le fragment de chemin de fer Name est représenté séparément du diagramme de chemin de fer principal.
Valeurs de ponctuation et de majuscules	Spécifiez exactement comme indiqué.
Valeurs en minuscules (par exemple, <i>name</i>)	Indiquez votre propre texte à la place de la variable <i>name</i> .

Commandes MFT de traçage

Vous pouvez tracer l'une des commandes Managed File Transfer pour vous aider à identifier les problèmes à partir de la ligne de commande.

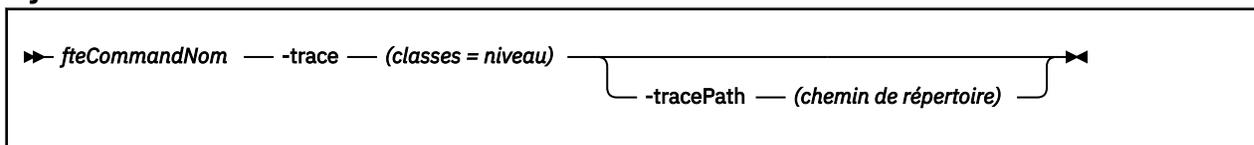
Objet

Utilisez le paramètre **-trace** pour toute commande MFT afin d'activer la trace à un niveau spécifié. Les fichiers de trace générés se trouvent dans votre répertoire de travail en cours, sauf si le paramètre **-tracePath** est inclus pour identifier un autre répertoire.

Etant donné que l'exécution de la trace peut affecter vos performances de manière significative et peut produire une grande quantité de données de trace, exécutez la trace avec précaution et uniquement lorsque cela est nécessaire. En règle générale, activez la fonction de trace uniquement lorsque le technicien de maintenance IBM vous le demande.

Vous pouvez définir d'autres propriétés de trace, par exemple la taille du fichier de trace et le nombre de fichiers de trace à conserver, dans le fichier `agent.properties`. Ces propriétés sont décrites dans [Propriétés avancées de l'agent](#).

Syntaxe



Paramètres

-trace (classes=niveau)

Obligatoire. Niveau de définition de la trace et des classes auxquelles la trace doit être appliquée. Indiquez le format suivant:

```
classes=level
```

Exemple :

```
com.ibm.wmqfte=all
```

qui trace toutes les classes Managed File Transfer .

Spécifiez une liste de spécifications de classe séparées par un signe deux-points auxquelles vous souhaitez appliquer le niveau de trace. Si vous ne spécifiez pas ce paramètre, le niveau de trace est appliqué à toutes les classes d'agent.

Si (*classes*) commencent par un signe plus (+), la liste des classes de trace qui suit le signe plus est ajoutée à toutes les classes de trace existantes actuellement tracées.

Les options de niveau de trace valides sont les suivantes et sont répertoriées par ordre croissant de taille et de détail du fichier de trace:

désactivé

Désactive la trace de l'agent mais continue d'écrire des informations dans les fichiers journaux. Il s'agit de l'option par défaut.

flux

Capture des données pour les points de trace associés au flux de traitement dans l'agent.

modéré

Capture une quantité modérée d'informations de diagnostic dans la trace.

prolix

Capture une quantité prolix d'informations de diagnostic dans la trace.

Tous

Définit la trace d'agent à exécuter sur toutes les classes d'agent.

-tracePath (chemin de répertoire)

Facultatif. Indiquez le répertoire dans lequel la trace doit être écrite. Par exemple, c : \temp.

 Si vous ne spécifiez pas ce paramètre, la valeur est le répertoire à partir duquel la commande a été émise. Par exemple, sous z/OS:

```
 /u/smith/fte/wmqmft/mqft/logs/MQPV/loggers/BFGLG1/logs/
```

Ce paramètre est valide uniquement lorsque le paramètre **-trace** est spécifié.

Exemple

Dans cet exemple, le niveau de trace est défini sur all, ce qui signifie que toutes les classes appartenant à AGENT.NAME sont tracés pour la commande **fteStartAgent** :

Remarque : Lorsque l'agent est démarré, la trace est soumise à *mft_config/logscoordination_qmgr/agents/agent*

```
fteStartAgent -trace com.ibm.wmqfte=all -tracePath /u/mft/trace AGENT.NAME
```

Dans cet exemple, le niveau de trace est défini sur modéré pour les classes com.ibm.wmqfte.common de l'agent AGENT.NAME. Une quantité modérée de trace est capturée pour la commande **ftePingAgent** :

```
ftePingAgent -trace com.ibm.wmqfte.common=moderate AGENT.NAME
```

Dans cet exemple, le niveau de trace est défini sur modéré pour les classes com.ibm.wmqfte.common de l'agent AGENT.NAME, et la trace est écrite dans le répertoire c:\\$user. Une quantité modérée de trace est capturée pour la commande **ftePingAgent** :

```
ftePingAgent -trace com.ibm.wmqfte.common=moderate -tracePath c:\$user AGENT.NAME
```

Référence associée

«traitement des incidentsMFT», à la page 277

Utilisez les informations de référence suivantes pour vous aider à diagnostiquer les erreurs dans Managed File Transfer:

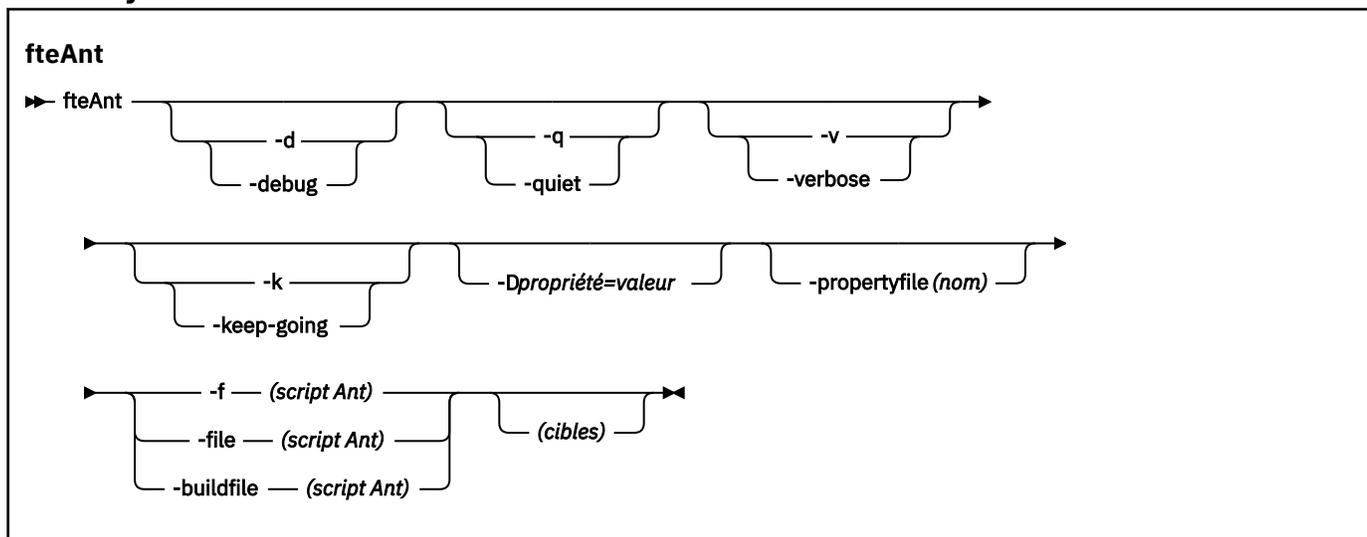
fteAnt: exécuter des tâches Ant dans MFT

La commande **fteAnt** exécute des scripts Ant dans un environnement dans lequel des tâches Ant Managed File Transfer sont disponibles.

Objet

Utilisez la commande **fteAnt** pour exécuter un script Ant dans un environnement avec Managed File Transfer. Contrairement à la commande **ant** standard, **fteAnt** requiert que vous définissiez un fichier script.

Syntaxe



Paramètres

-debug ou -d

Facultatif. Générez une sortie de débogage.

-quiet ou -q

Facultatif. Générer une sortie minimale.

-verbose ou -v

Facultatif. Générer une sortie prolixe.

-continue ou -k

Facultatif. Exécuter toutes les cibles qui ne dépendent pas des cibles ayant échoué.

-D propriété=valeur

Facultatif. Utilisez *value* pour une propriété *donnée*. Les propriétés définies avec **-D** sont prioritaires par rapport à celles définies dans un fichier de propriétés.

Utilisez la propriété **com.ibm.wmqfte.propertyset** pour spécifier l'ensemble des options de configuration utilisées pour les tâches Ant. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme valeur de cette propriété. Les tâches Ant utilisent ensuite l'ensemble des options de configuration associées à ce gestionnaire de files d'attente de coordination autre que celui par défaut. Si vous ne spécifiez pas cette propriété, l'ensemble d'options de configuration par défaut basé sur le gestionnaire de files d'attente de coordination par défaut est utilisé. Si vous spécifiez l'attribut **cmdqm** pour une tâche Ant, cet attribut est prioritaire sur l'ensemble des options de configuration spécifiées pour la commande **fteAnt**. Ce comportement s'applique que vous utilisiez l'ensemble d'options de configuration par défaut ou que vous spécifiiez un ensemble avec la propriété **com.ibm.wmqfte.propertyset**.

-propertyfile (nom)

Facultatif. Chargez toutes les propriétés à partir d'un fichier dont les propriétés **-D** sont prioritaires.

-f (script Ant), -file (script Ant) ou -buildfile (script Ant)

Obligatoire. Indique le nom du script Ant à exécuter.

cibles

Facultatif. Nom d'une ou de plusieurs cibles à exécuter à partir du script Ant. Si vous ne spécifiez pas de valeur pour ce paramètre, la cible par défaut du script est exécutée.

-version

Facultatif. Affiche la commande Managed File Transfer et les versions Ant.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, la cible **copy** dans le script Ant `fte_script.xml` est exécutée et la commande écrit la sortie de débogage dans la sortie standard.

```
fteAnt -d -f fte_script.xml copy
```

Codes retour**0**

La commande a abouti.

1

La commande a échoué.

D'autres codes retour de statut peuvent également être spécifiés à partir de scripts Ant, par exemple à l'aide de la tâche d'échec Ant.

Pour plus d'informations, voir [Echec](#).

z/OS fteBatch, fteCommon et ftePlatform

z/OS fteBatch, fteCommon et ftePlatform sont des scripts fournis par Managed File Transfer dans le répertoire `MQ_INSTALLATION_PATH/bin` en tant que scripts auxiliaires. Tous ces scripts ne sont pas présents sur chaque plateforme.

Script fteBatch (z/OS uniquement)**z/OS**

fteBatch est un script auxiliaire permettant d'exécuter Managed File Transfer à partir de JZOS Batch Launcher. fteBatch est installé sur z/OS uniquement. En règle générale, Managed File Transfer est démarré à l'aide des scripts de shell de commandes fournis, qui effectuent une configuration d'environnement avant de démarrer la classe Java appropriée à cette fonction. Lorsque Managed File Transfer est démarré à l'aide de JZOS Batch Launcher, la classe Java est démarrée directement à partir de Launcher. fteBatch peut être appelé dans le cadre de la configuration du lanceur pour placer le nom de classe requis dans une variable d'environnement et effectuer le travail de configuration effectué par les scripts de shell de commandes normaux avant de démarrer Java. Cela fournit un niveau d'isolement entre vos travaux et les noms de classe internes utilisés par Managed File Transfer.

La commande fteBatch est obsolète pour Managed File Transfer 8.0, car vous pouvez exécuter Managed File Transfer via le nouveau fichier de commandes PDSE. Pour plus d'informations, voir [«Création d'un fichier de commandes d'agent ou de consignateur MFT»](#), à la page 64.

fteCommon

fteCommon est un script auxiliaire démarré par les autres scripts de commande Managed File Transfer pour effectuer un traitement de configuration commun avant de démarrer Java.

ftePlatform

ftePlatform est un script auxiliaire démarré par le script fteCommon pour effectuer un traitement de configuration spécifique à la plateforme.

fteCancelTransfer

La commande **fteCancelTransfer** permet d'annuler un transfert Managed File Transfer . Vous pouvez exécuter cette commande sur l'agent source ou de destination pour le transfert.

Objet

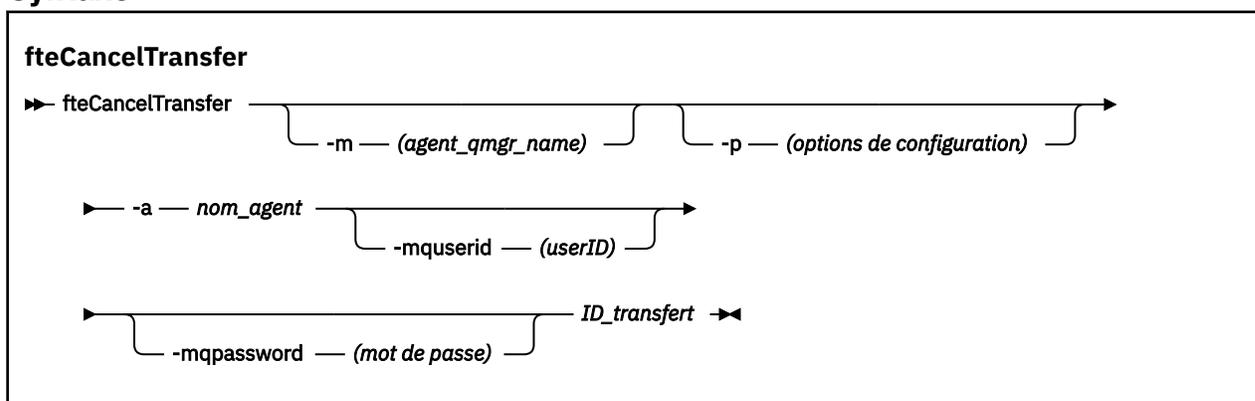
Si vous exécutez la commande **fteCancelTransfer** alors que ce transfert est en cours, tous les fichiers déjà transférés dans le cadre de ce transfert restent sur le système cible et ne sont pas supprimés. Tous les fichiers partiellement transférés dans le cadre de ce transfert sont supprimés du système de destination. Côté destination des journaux de transfert qui sont transférés en tant que "annulés".

Si un transfert vers un noeud Connect:Direct est annulé, tous les fichiers partiellement transférés dans le cadre du transfert annulé restent sur le système cible et ne sont pas supprimés.

Vous pouvez exécuter la commande **fteCancelTransfer** à partir de n'importe quel système pouvant se connecter au réseau IBM MQ , puis effectuer le routage vers le gestionnaire de files d'attente d'agent. Pour que la commande puisse être exécutée, vous devez avoir installé Managed File Transfer sur ce système et avoir configuré Managed File Transfer sur ce système pour communiquer avec le réseau IBM MQ . Si aucun détail de connectivité n'est disponible, les détails du gestionnaire de files d'attente de l'agent sont utilisés à la place pour la connexion, à condition que ces détails soient disponibles.

Spécifiez le paramètre facultatif **-p** pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de votre ensemble par défaut. Pour plus d'informations, voir [Options de configuration](#) .

Syntaxe



Paramètres

-m (agent_qmgr_name)

Facultatif. Nom du gestionnaire de files d'attente d'agent. Cet agent doit être l'agent source ou de destination du transfert que vous souhaitez annuler. Si vous ne spécifiez pas ce paramètre, la demande d'annulation est envoyée au gestionnaire de files d'attente identifié par l'ensemble d'options de configuration que vous utilisez.

-p (options_configuration)

Facultatif. Ce paramètre détermine l'ensemble des options de configuration à utiliser pour annuler le transfert. Par convention, utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

-a (nom_agent)

Obligatoire. Nom de l'agent source ou de destination du transfert que vous souhaitez annuler.

-mquserid (userID)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de files d'attente de commandes.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de commandes. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

ID_transfert

Obligatoire. ID du transfert que vous souhaitez annuler. L'ID de transfert (également appelé ID de demande) s'affiche sur la ligne de commande après l'exécution de la commande **fteCreateTransfer**. Les ID de transfert sont également inclus dans les messages du journal de transfert de fichier ou sont affichés dans le panneau IBM MQ Journal de transfert de l'explorateur.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, AGENT1 est l'agent source du transfert à annuler.

```
fteCancelTransfer -a AGENT1 414d5120514d5f4c4d343336303920201159c54820027102
```

Codes retour

0

La commande a abouti ou l'ID de transfert spécifié est inconnu de l'agent. Si l'ID de transfert est inconnu de l'agent, la raison la plus probable est que le transfert est déjà terminé ou qu'il a été annulé.

1

La commande a échoué.

Référence associée

«[fteCreateTransfer: démarrer un nouveau transfert de fichier](#)», à la page 447

La commande **fteCreateTransfer** crée et démarre un nouveau transfert de fichier à partir de la ligne de commande. Cette commande peut démarrer un transfert de fichier immédiatement, planifier un transfert de fichier à une heure et une date ultérieures, répéter un transfert planifié une ou plusieurs fois et déclencher un transfert de fichiers sous certaines conditions.

fteChangeDefaultConfigurationOptions

Utilisez la commande **fteChangeDefaultConfigurationOptions** pour modifier les options de configuration par défaut que Managed File Transfer doit utiliser. La valeur des options de configuration définit le groupe de fichiers de propriétés utilisé par Managed File Transfer .

Important :  Sous IBM MQ for UNIX, Linux, and Windows, seuls les utilisateurs qui sont des administrateurs d'IBM MQ (et membres du groupe mqm) peuvent exécuter cette commande. Si vous tentez d'exécuter cette commande en tant qu'utilisateur qui n'est pas un administrateur IBM MQ , le message d'erreur BFGCL0502E: Vous n'êtes pas autorisé à effectuer l'opération demandée . et la commande n'est pas exécutée.

 Sur les systèmes z/OS, l'utilisateur doit remplir (au moins) l'une des conditions suivantes pour pouvoir exécuter la commande migrate :

- Etre membre du groupe mqm (si le groupe mqm est défini sur le système).
-  Etre membre du groupe désigné dans la variable d'environnement BFG_GROUP_NAME (si un tel groupe a été désigné).
-  Ne pas avoir de valeur définie dans la variable d'environnement BFG_GROUP_NAME lorsque la commande est exécutée.

Objet

Vos options de configuration Managed File Transfer par défaut sont établies la première fois que vous utilisez la commande **fteSetupCoordination** pour configurer un gestionnaire de files d'attente en tant que gestionnaire de files d'attente de coordination. Lors de l'installation du produit MFT , le répertoire mqft est créé sous <MQ_DATA_PATH> s'il n'existe pas déjà. De plus, les répertoires de configuration, d'installation et de journaux sont créés sous le répertoire mqft , s'ils n'existent pas déjà.

A l'aide de la commande **fteChangeDefaultConfigurationOptions** , vous pouvez modifier le gestionnaire de files d'attente de coordination par défaut défini dans le fichier `installation.properties` . Si vous modifiez ce gestionnaire de files d'attente de coordination, Managed File Transfer utilise les options de configuration fournies par l'ensemble structuré de répertoires et de fichiers de propriétés qu'il contient, le répertoire que vous avez utilisé comme entrée pour `options_configuration` par défaut. Ce nom de répertoire est identique au gestionnaire de files d'attente de coordination utilisé par les agents dans cette configuration.

Voir «Options de configuration de MFT sur Multiplatforms», à la page 58 pour plus d'informations sur le fichier `installation.properties` .

Syntaxe

fteChangeDefaultConfigurationOptions

►► fteChangeDefaultConfigurationOptions — *options_configuration* ◄◄

Paramètres

options_configuration

Obligatoire. Ce paramètre indique les options de configuration par défaut que vous souhaitez modifier. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, les options de configuration par défaut sont remplacées par QM_COORD2:

```
fteChangeDefaultConfigurationOptions QM_COORD2
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Concepts associés

«Options de configuration de MFT sur Multiplatforms», à la page 58

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.

Agent fteClean: nettoyage d'un agent MFT

La commande **fteCleanAgent** permet de nettoyer les files d'attente utilisées par un Managed File Transfer Agent en supprimant les messages des files d'attente persistantes et non persistantes utilisées par l'agent. Utilisez la commande **fteCleanAgent** si vous rencontrez des problèmes lors du démarrage d'un agent, ce qui peut être dû aux informations restant dans les files d'attente utilisées par l'agent.

Objet

La commande **fteCleanAgent** permet de supprimer des messages des files d'attente persistantes et non persistantes utilisées par l'agent. Plus précisément, cette commande peut effectuer les actions suivantes:

- Supprimez tous les transferts en cours vers cet agent ou à partir de cet agent avant l'arrêt du transfert. Ces transferts ne sont pas repris lorsque l'agent redémarre
- Supprimez toutes les commandes qui ont déjà été soumises à l'agent, mais qui n'ont pas encore été exécutées
- Supprimer tous les moniteurs de ressources stockés sur l'agent
- Supprimer tous les transferts planifiés stockés sur l'agent
- Supprimer tous les messages non valides stockés sur l'agent

Si l'agent est un agent de pont Connect:Direct, les paramètres **-ms**, **-sset** et **-ims** ne sont pas valides. Pour les agents de pont Connect:Direct, la commande effectue également les actions suivantes:

- Supprime tous les fichiers du répertoire dans lequel l'agent de pont Connect:Direct stocke temporairement les fichiers lors de leur transfert. L'emplacement de ce répertoire est défini par le paramètre **cdTmpDir**
- Affiche des informations sur les processus Connect:Direct associés à des transferts en cours

Avant IBM MQ 9.0.3 et IBM MQ 9.0.0 Fix Pack 2, si vous exécutez la commande **fteCleanAgent** avec uniquement le paramètre **agent_name**, la commande s'exécute par défaut comme si le paramètre **-all** avait été fourni, ce qui entraîne la suppression de tous les transferts planifiés, du moniteur de ressources et des définitions de transfert planifiées sur l'agent.

V 9.0.3 ➔ **V 9.0.0.2** Depuis IBM MQ 9.0.3 et IBM MQ 9.0.0 Fix Pack 2, vous devez, par défaut, spécifier l'état Managed File Transfer à effacer en transmettant les paramètres appropriés à la commande **fteCleanAgent** et en fournissant un nom d'agent. Cela signifie que, par défaut, **fteCleanAgent** n'efface pas tous les transferts en cours et en attente, les définitions de moniteur de ressources et les définitions de transfert planifiées pour l'agent spécifié. Vous pouvez activer ou

désactiver ce comportement en définissant la propriété `failCleanAgentWithNoArguments` du fichier `command.properties` sur la valeur appropriée:

- Par défaut, la valeur de `failCleanAgentWithNoArguments` est `true`, ce qui signifie que l'exécution de la commande **fteCleanAgent** échoue si seul le paramètre **agent_name** est spécifié.
- Si `failCleanAgentWithNoArguments` est défini sur `false` et que seul le paramètre **agent_name** est spécifié, **fteCleanAgent** se comporte de la même manière que lorsque vous spécifiez le paramètre **-all**.

Vous devez exécuter la commande **fteCleanAgent** sur un agent qui a été arrêté. Si vous tentez d'exécuter la commande sur un agent en cours d'exécution, vous recevez une erreur. Cette commande ne démarre pas l'agent. La commande **fteCleanAgent** nettoie un agent sur le système sur lequel vous émettez la commande. Vous ne pouvez pas nettoyer un agent sur un système distant. Pour exécuter la commande **fteCleanAgent**, vous devez disposer d'un accès en écriture au fichier de verrouillage de l'agent, qui se trouve dans `MQ_DATA_PATH\mqft\logs\coordination_QMgr_name\agents\agent_name\agent.lck`

Le groupe FTEAGENT doit disposer des droits GET et BROWSE sur les files d'attente suivantes pour que l'exécution de **fteCleanAgent** aboutisse:

- SYSTEME SYSTEM.FTE.COMMAND.*nom_agent*
- SYSTEME SYSTEM.FTE.EVENT.*nom_agent*
- SYSTEME SYSTEM.FTE.STATE.*nom_agent*

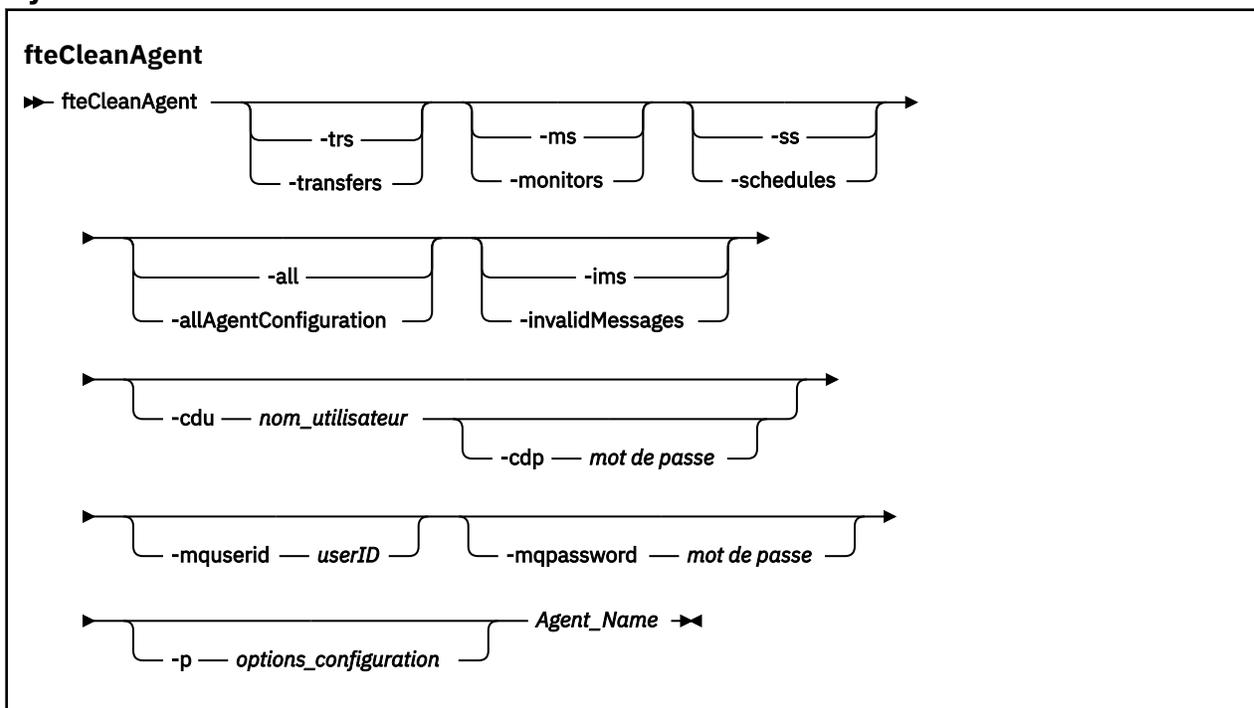
Voir «Restriction des droits de groupe pour les ressources spécifiques à MFT», à la page 362 pour plus d'informations sur le groupe FTEAGENT et la restriction des droits de groupe.

Si vous exécutez la commande **fteCleanAgent** sur un agent connecté à son gestionnaire de files d'attente en mode liaisons et que l'agent s'est arrêté récemment, la commande **fteCleanAgent** peut signaler un problème de messagerie: MQRC 2042. Ce MQRC se produit car un descripteur de file d'attente pour l'agent existe toujours dans le gestionnaire de files d'attente. Après un bref délai, le gestionnaire de files d'attente supprime cet identificateur et vous pouvez réexécuter **fteCleanAgent**.

Spécifiez le paramètre facultatif **-p** pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de votre ensemble par défaut. Pour plus d'informations, voir [Options de configuration](#).

Remarque : Lors du nettoyage d'un agent de pont Connect:Direct, l'ID utilisateur utilisé pour exécuter la commande **fteCleanAgent** doit disposer d'un accès en lecture et en écriture au répertoire temporaire de l'agent de pont Connect:Direct.

Syntaxe



Paramètres

Vous pouvez utiliser la commande **fteCleanAgent** pour supprimer des artefacts spécifiques. Par exemple, vous pouvez spécifier la commande **-trs** pour supprimer les transferts en attente mais ne pas modifier les moniteurs de ressources et les transferts planifiés.

-trs ou **-transferts**

Facultatif. Indique que les transferts en cours et en attente doivent être supprimés de l'agent. Vous ne pouvez pas spécifier ce paramètre avec les paramètres **-all** ou **-ims**.

-ms ou **-monitors**

Facultatif. Indique que toutes les définitions de moniteur de ressources doivent être supprimées de l'agent. Vous ne pouvez pas spécifier ce paramètre avec les paramètres **-all** ou **-ims**.

-ss ou **-plannings**

Facultatif. Indique que toutes les définitions de transfert planifiées doivent être supprimées de l'agent. Vous ne pouvez pas spécifier ce paramètre avec les paramètres **-all** ou **-ims**.

-all ou **-allAgentConfiguration**

Facultatif. Indique que tous les transferts, les définitions de moniteur de ressources et les définitions de transfert planifiées doivent être supprimés de l'agent. Vous ne pouvez pas spécifier ce paramètre avec les paramètres **-trs**, **-ss**, **-ms** ou **-ims**.



Avertissement : Vous ne devez utiliser le paramètre **all** que si aucune autre option n'est disponible. La suppression de transferts, de définitions de moniteur de ressources et de définitions de transfert planifiées peut avoir un impact significatif sur votre entreprise.

-ims ou **-invalidMessages**

Facultatif. Indique que tous les messages non valides doivent être supprimés de l'agent. Vous ne pouvez pas spécifier ce paramètre avec les paramètres **-trs**, **-ss**, **-ms** ou **-all**.

-cdu *nom_utilisateur*

Facultatif. Valide uniquement si l'agent en cours de nettoyage est un agent de pont Connect:Direct. Si ce paramètre est spécifié, la commande utilise le nom d'utilisateur fourni pour établir une

Le fichier `command.properties` indique le gestionnaire de files d'attente de commandes auquel vous devez vous connecter lorsque vous émettez des commandes et les informations dont Managed File Transfer a besoin pour contacter ce gestionnaire de files d'attente.

Agent `fteCreate`(créer un agent MFT)

La commande `fteCreateAgent` crée un Managed File Transfer Agent et sa configuration associée.

Vous pouvez contrôler l'accès à l'agent. Pour plus d'informations, voir [Restriction des droits utilisateur sur les actions de l'agent MFT](#) . Vous devez utiliser le paramètre `-ac` et accorder des droits d'accès à certaines files d'attente.

Important : `ULW` Sous IBM MQ for UNIX, Linux, and Windows, seuls les utilisateurs qui sont des administrateurs d'IBM MQ (et membres du groupe `mqm`) peuvent exécuter cette commande. Si vous tentez d'exécuter cette commande en tant qu'utilisateur qui n'est pas un administrateur IBM MQ, le message d'erreur BFGCL0502E: Vous n'êtes pas autorisé à effectuer l'opération demandée. et la commande n'est pas exécutée.

`z/OS` Sur les systèmes z/OS, l'utilisateur doit remplir (au moins) l'une des conditions suivantes pour pouvoir exécuter la commande `migrate` :

- Etre membre du groupe `mqm` (si le groupe `mqm` est défini sur le système).
- `V9.0.0.1` Etre membre du groupe désigné dans la variable d'environnement `BFG_GROUP_NAME` (si un tel groupe a été désigné).
- `V9.0.0.1` Ne pas avoir de valeur définie dans la variable d'environnement `BFG_GROUP_NAME` lorsque la commande est exécutée.

Objet

Utilisez la commande `fteCreateAgent` pour créer un agent. Cette commande fournit les commandes MQSC que vous devez exécuter sur votre gestionnaire de files d'attente d'agent pour créer les files d'attente d'agent suivantes:

- `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`
- `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`

Ces files d'attente sont des files d'attente système internes à partir de laquelle vous ne devez pas modifier, supprimer ou lire des messages, sauf si vous supprimez l'agent. Les commandes MQSC à exécuter sont également fournies dans un fichier à l'emplacement suivant:

```
MQ_DATA_PATH\mqft\config\coordination_qmgr_name\agents\agent_name\agent_name_create.mqsc.
```

Si vous souhaitez supprimer ultérieurement l'agent, cette commande fournit également les commandes MQSC que vous devez exécuter pour effacer puis supprimer les files d'attente utilisées par l'agent. Les commandes MQSC se trouvent dans un fichier à l'emplacement suivant:

```
MQ_DATA_PATH\mqft\config\coordination_qmgr_name\agents\agent_name\agent_name_delete.mqsc.
```

Managed File Transfer fournit des propriétés d'agent avancées qui vous aident à configurer des agents. Ces propriétés sont décrites dans le [fichier agent.properties](#).

Il se peut que vous deviez créer un fichier de données d'identification `MQMFTCredentials.xml` pour pouvoir utiliser votre agent. Un exemple de ce fichier se trouve dans `MQ_INSTALLATION_PATH/mqft/samples/credentials/`. Pour plus d'informations et d'exemples, consultez «[Format de fichier de données d'identification MFT](#)», à la page 835.

Important :

Sur les plateformes UNIX et les commandes Linux Managed File Transfer , utilisez des fichiers de socket pour communiquer avec le processus d'agent exécuté sur la même machine hôte.

Ces fichiers de socket sont créés dans le répertoire de journaux de l'agent et sont supprimés lorsqu'un agent s'arrête. Dans l'installation de IBM MQ Managed File Transfer , ce fichier de socket est créé avec le chemin d'accès `<MQ_DATA_PATH>/mqft/logs/<COORDINATION_QM_NAME>/agents/<AGENT_NAME>/logs/<AGENT_NAME>@<AGENT_QM_NAME>` où `MQ_DATA_PATH` est `/var/mqm` par défaut.

Pour un agent redistribuable, ce fichier de socket est créé sous le répertoire: `<RE_DISTRIBUTABLE_DIRECTORY>/mqft/logs/<COORDINATION_QM_NAME>/agents/<AGENT_NAME>/logs/<AGENT_NAME>@<AGENT_QM_NAME>`.

Par exemple, si le nom de l'agent est SRCAGENT, que le nom du gestionnaire de files d'attente de l'agent est SRCAGENTQM, que le nom du gestionnaire de files d'attente de coordination est COORDQM et que l'agent redistribuable s'exécute à partir du répertoire `/home/myuser/mqmft-redis`, le chemin d'accès complet de ce fichier de socket est: `/home/myuser/mqmft-redis/mqft/logs/COORDQM/agents/SRCAGENT/logs/SRCAGENT@SRCAGENTQM`

qui correspond à une longueur totale de chemin de fichier de 85 caractères.

La longueur de chemin maximale autorisée par ces systèmes d'exploitation pour un fichier socket est de 107 caractères. Par conséquent, lors de la création d'un agent, veillez à ce que le chemin du fichier de socket ne dépasse pas 107 caractères. Ceci est particulièrement important avec un agent redistribuable dans lequel le répertoire de journaux de l'agent peut se trouver dans un emplacement de répertoire arbitraire. Pour plus de détails sur la configuration du répertoire de configuration, voir la commande **[fteCreateEnvironment](#)**.

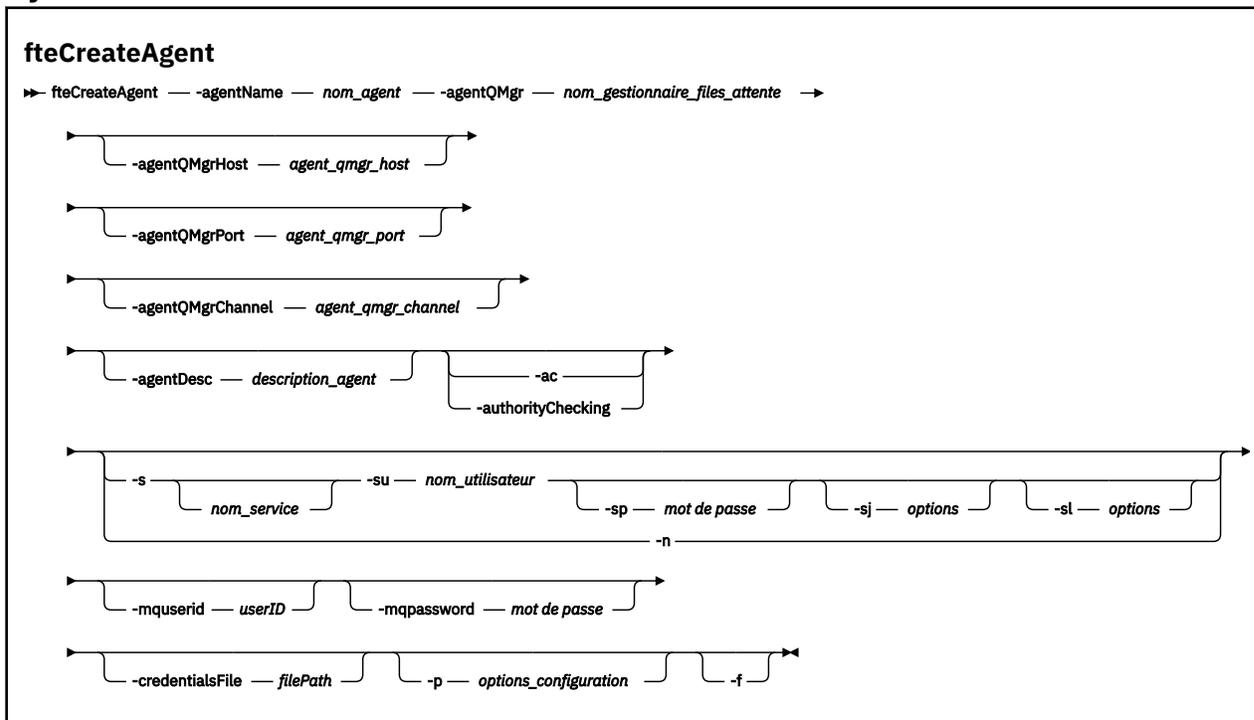
Si vous démarrez un agent ou que d'autres commandes qui se connectent à l'agent sont exécutées et que la longueur de votre chemin dépasse 107 caractères, vous recevez le message suivant:

```
BFGNV0159E: Echec de la tentative de liaison au fichier de socket avec FFDC
```

Caractères spéciaux

Soyez attentif lorsque vous utilisez des valeurs de paramètre contenant des caractères spéciaux pour que l'interpréteur de commandes interprète les caractères de façon appropriée. Par exemple, les noms et les chemins de fichier qualifiés complets contenant des caractères tels qu'un espace, une apostrophe ou des guillemets, ou bien une barre oblique ou une barre oblique inversée peuvent être interprétés par l'interpréteur de commandes au lieu d'être transmis directement à la commande. Pour éviter que des caractères soient interprétés par l'interpréteur de commandes, placez l'intégralité du paramètre entre apostrophes ou guillemets ou associez les caractères spéciaux à la séquence d'échappement de l'interpréteur de commandes.

Syntaxe



Paramètres

-agentName (*nom_agent*)

Obligatoire. Nom de l'agent que vous souhaitez créer. Le nom de l'agent doit être unique dans son gestionnaire de files d'attente de coordination.

Pour plus d'informations sur les agents de dénomination, voir [Conventions de dénomination d'objet](#).

-agentQMGr (*agent_qmgr_name*)

Obligatoire. Nom du gestionnaire de files d'attente d'agent.

-agentQMGrHôte (*agent_qmgr_host*)

Facultatif. Nom d'hôte ou adresse IP du gestionnaire de files d'attente de l'agent.

-agentQMGrPort (*agent_qmgr_port*)

Facultatif. Numéro de port utilisé pour les connexions client au gestionnaire de files d'attente d'agent.

-agentQMGrCanal (*agent_qmgr_channel*)

Facultatif. Nom de canal utilisé pour la connexion au gestionnaire de files d'attente d'agent.

-agentDesc (*description_agent*)

Facultatif. Description de l'agent, qui s'affiche dans IBM MQ Explorer.

-ac ou -authorityChecking

Facultatif. Ce paramètre permet le contrôle des droits d'accès. Si vous spécifiez ce paramètre, l'agent vérifie que les utilisateurs qui soumettent des demandes sont autorisés à effectuer l'action demandée. Pour plus d'informations, voir [Restriction des droits utilisateur sur les actions de l'agent MFT](#).

Windows

-s (*nom_service*)

Facultatif (Windows uniquement). Indique que l'agent doit être exécuté en tant que service Windows, la commande doit être exécutée à partir d'un ID administrateur Windows. Si vous ne spécifiez pas *service_name*, le service est nommé mqmftAgentAGENTQMGR, où *AGENT* est le nom de l'agent et *QMGR* est le nom de votre gestionnaire de files d'attente d'agent.

Le nom d'affichage du service, qui est affiché dans la fenêtre Windows **Services** de la colonne **Nom** , est toujours **Managed File Transfer Agent AGENT@QMGR**.

Remarque : Si l'agent redistribuable doit s'exécuter en tant que service Windows, vous devez définir la variable d'environnement **BFG_DATA** dans l'environnement système pour que le service fonctionne.

Windows -su (nom_utilisateur)

Facultatif (Windows uniquement). Lorsque l'agent doit s'exécuter en tant que service Windows , ce paramètre indique le nom du compte sous lequel le service s'exécute. Pour exécuter l'agent à l'aide d'un compte utilisateur de domaine Windows , indiquez la valeur dans le formulaire `DomainName\UserName`. Pour exécuter le service à l'aide d'un compte à partir du domaine intégré local, spécifiez la valeur dans le formulaire `UserName`.

Le compte utilisateur Windows que vous spécifiez à l'aide du paramètre **-su** doit disposer du droit **Log on as a service** . Pour plus d'informations sur l'octroi de ce droit, voir [Conseils pour l'exécution d'un agent ou d'un consignateur MFT en tant que service Windows](#).

Obligatoire lorsque **-s** est spécifié.

Windows -sp (mot_de_passe)

Facultatif (Windows uniquement).

Ce paramètre est valide uniquement lorsque **-s** est spécifié. Si vous ne spécifiez pas ce paramètre lorsque vous spécifiez le paramètre **-s** , un message d'avertissement est généré. Ce message vous avertit que vous devez définir le mot de passe à l'aide de l'outil Windows Services avant que le service ne démarre correctement.

Windows -sj (options)

Facultatif (Windows uniquement). Lorsque l'agent est démarré en tant que service Windows , définit une liste d'options sous la forme `-D` ou `-X` qui sont transmises à la machine virtuelle Java. Les options sont séparées par un signe dièse (`#`) ou un point-virgule (`;`). Si vous devez incorporer des caractères `#` ou point-virgule (`;`), placez-les entre apostrophes.

Ce paramètre est valide uniquement lorsque **-s** est spécifié.

Windows -sl (options)

Facultatif (Windows uniquement). Définit le niveau de journalisation du service Windows . Les options valides sont: `error`, `info`, `warn`, `debug`. La valeur par défaut est `info`. Cette option peut être utile si vous rencontrez des problèmes avec le service Windows . Le fait de le définir sur `debug` fournit des informations plus détaillées dans le fichier journal du service.

Ce paramètre est valide uniquement lorsque **-s** est spécifié.

Windows -n

Facultatif (Windows uniquement). Indique que l'agent doit être exécuté en tant que processus normal. Cette option est mutuellement exclusive avec l'option **-s** . Si aucun des paramètres **-s** et **-n** n'est spécifié, l'agent est configuré en tant que processus Windows normal.

-p (options_configuration)

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour créer un agent. Par convention, utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande **fteCreateAgent** utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Spécifiez le paramètre facultatif **-p** uniquement si vous souhaitez utiliser des options de configuration différentes de vos valeurs par défaut. Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

-mquserid (userID)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de file d'attente de coordination.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de coordination. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

-credentialsFile (filePath)

Facultatif. Chemin d'accès complet à un fichier de données d'identification existant ou nouveau, dans lequel les détails de l'authentification IBM MQ sont ajoutés.

Cette commande prend en charge l'ajout d'un ensemble de détails d'authentification IBM MQ à un fichier de données d'identification Managed File Transfer nommé. Utilisez-la si l'authentification de connexion IBM MQ a été activée. Si vous mettez à jour les détails existants, vous devez utiliser le paramètre force **-f**.

-credentialPath (chemin_données).

Cette commande définit l'emplacement vers lequel les données d'identification doivent être migrées. Ce paramètre peut être un chemin de répertoire vers un fichier de données d'identification existant ou un chemin de répertoire vers un nouveau fichier de données d'identification.  Sur les plateformes z/OS, le fichier de données d'identification peut être un ensemble de données partitionnées étendu (PDSE) préexistant. Le PDSE peut inclure des membres existants ou un nouveau membre pour le fichier de données d'identification. Les membres existants du PDSE doivent être mis à jour pour inclure le fichier de données d'identification. Le format de l'ensemble de données partitionnées étendu doit être bloqué.

-f

Facultatif. Force la commande à remplacer les paramètres existants qui ne correspondent pas. La spécification de ce paramètre ne force pas le remplacement d'un agent de service Windows existant.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, AGENT3 est créé avec un gestionnaire de files d'attente d'agent QM_NEPTUNE et utilise le gestionnaire de files d'attente de coordination par défaut:

```
fteCreateAgent -agentName AGENT3 -agentQMGr QM_NEPTUNE
-agentQMGrHost myhost.ibm.com -agentQMGrPort 1415 -agentQMGrChannel CHANNEL1
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

fteCreateBridgeAgent (créer et configurer un agent de pont de protocole MFT)

La commande **fteCreateBridgeAgent** crée un agent de pont de protocole Managed File Transfer et sa configuration associée. Créez un agent de pont de protocole pour chaque serveur de fichiers auquel vous souhaitez envoyer des fichiers et duquel vous souhaitez les recevoir.

Important :  Sous IBM MQ for UNIX, Linux, and Windows, seuls les utilisateurs qui sont des administrateurs d'IBM MQ (et membres du groupe mqm) peuvent exécuter cette commande. Si vous tentez d'exécuter cette commande en tant qu'utilisateur qui n'est pas un administrateur IBM MQ, le message d'erreur BFGCL0502E: Vous n'êtes pas autorisé à effectuer l'opération demandée. et la commande n'est pas exécutée.

z/OS Sur les systèmes z/OS, l'utilisateur doit remplir (au moins) l'une des conditions suivantes pour pouvoir exécuter la commande `migrate` :

- Etre membre du groupe `mqm` (si le groupe `mqm` est défini sur le système).
- **V9.0.0.1** Etre membre du groupe désigné dans la variable d'environnement `BFG_GROUP_NAME` (si un tel groupe a été désigné).
- **V9.0.0.1** Ne pas avoir de valeur définie dans la variable d'environnement `BFG_GROUP_NAME` lorsque la commande est exécutée.

Objet

La commande **fteCreateBridgeAgent** permet de créer un agent de pont de protocole. Pour une présentation de l'utilisation du pont de protocole, voir [Le pont de protocole](#). Cette commande **fteCreateBridgeAgent** fournit les commandes MQSC que vous devez exécuter sur votre gestionnaire de files d'attente d'agent pour créer les files d'attente d'agent suivantes:

- `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`
- `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`

Ces files d'attente sont des files d'attente système internes à partir de laquelle vous ne devez pas modifier, supprimer ou lire des messages, sauf si vous supprimez l'agent. Les commandes MQSC à exécuter sont également fournies dans un fichier à l'emplacement suivant:

`MQ_DATA_PATH\mqft\config\coordination_qmgr_name\agents\agent_name\agent_name_create.mqsc`

Si vous souhaitez supprimer ultérieurement l'agent, cette commande fournit également les commandes MQSC que vous devez exécuter pour effacer, puis supprimer les files d'attente utilisées par l'agent. Les commandes MQSC se trouvent dans un fichier à l'emplacement suivant:

`MQ_DATA_PATH\mqft\config\coordination_qmgr_name\agents\agent_name\agent_name_delete.mqsc`

La commande **fteCreateBridgeAgent** crée un fichier XML `ProtocolBridgeProperties.xml` dans le répertoire suivant:
`MQ_DATA_PATH\mqft\config\coordination_qmgr_name\agents\agent_name`.

Les utilisateurs sont responsables de la création manuelle du fichier `ProtocolBridgeCredentials.xml`, qui n'est plus créé par la commande **fteCreateBridgeAgent**.

Le fichier `ProtocolBridgeCredentials.xml` vous permet de définir des noms d'utilisateur et des données d'identification que l'agent de pont de protocole utilise pour s'autoriser avec le serveur de protocole et le fichier `ProtocolBridgeProperties.xml` vous permet de définir plusieurs serveurs de fichiers de protocole afin de pouvoir effectuer un transfert vers plusieurs noeuds finaux.

Un exemple de fichier `ProtocolBridgeCredentials.xml` se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/credentials/`. Pour plus d'informations, voir «[Format de fichier des données d'identification du pont de protocole](#)», à la page 589 et «[Format du fichier de propriétés du pont de protocole](#)», à la page 592.

Si vous exécutez la commande **fteCreateBridgeAgent** et spécifiez un serveur de fichiers de protocole par défaut (paramètre -bt), ce serveur par défaut est contenu dans le fichier `ProtocolBridgeProperties.xml` et son nom d'hôte est utilisé comme nom de serveur. Avec le paramètre -bt, vous devez spécifier les paramètres suivants:

- -bh
- -btz
- -bm
- -bsl
- -bfe
- -bts

Si vous ne spécifiez pas de serveur par défaut, le fichier `ProtocolBridgeProperties.xml` ne contient aucune entrée ; vous devez ajouter au moins un serveur manuellement pour que les transferts puissent avoir lieu.

Managed File Transfer fournit des propriétés d'agent avancées qui vous aident à configurer des agents de pont de protocole. Les propriétés relatives au pont de protocole commencent par `protocol`. Ces propriétés sont décrites dans [Propriétés d'agent avancées: Pont de protocole](#) et [Propriétés d'agent avancées: Journalisation d'agent de pont de protocole](#). Si vous constatez un comportement inattendu dans le pont de protocole, passez en revue ces propriétés `protocol` et vérifiez que vous avez correctement défini ces propriétés pour votre système.

Si vous voyez la sortie suivante de la commande **fteCreateBridgeAgent** :

```
BFGMQ1007I: The coordination queue manager cannot be contacted or has refused a connection attempt.
The WebSphere MQ reason code was 2058. The agent's presence will not be published.
```

elle indique que le gestionnaire de files d'attente de coordination ne peut pas être contacté et fournit le code anomalie IBM MQ pour la raison. Ce message d'information peut indiquer que le gestionnaire de files d'attente de coordination est actuellement indisponible ou que vous avez défini la configuration de manière incorrecte.

Important :

Sur les plateformes UNIX et les commandes Linux Managed File Transfer, utilisez des fichiers de socket pour communiquer avec le processus d'agent exécuté sur la même machine hôte.

Ces fichiers de socket sont créés dans le répertoire de journaux de l'agent et sont supprimés lorsqu'un agent s'arrête. Dans l'installation de IBM MQ Managed File Transfer, ce fichier de socket est créé avec le chemin d'accès `<MQ_DATA_PATH>/mqft/logs/<COORDINATION_QM_NAME>/agents/<AGENT_NAME>/logs/<AGENT_NAME>@<AGENT_QM_NAME>` où `MQ_DATA_PATH` est `/var/mqm` par défaut.

Pour un agent redistribuable, ce fichier de socket est créé sous le répertoire: `<RE_DISTRIBUTABLE_DIRECTORY>/mqft/logs/<COORDINATION_QM_NAME>/agents/<AGENT_NAME>/logs/<AGENT_NAME>@<AGENT_QM_NAME>`.

Par exemple, si le nom de l'agent est `SRCAGENT`, que le nom du gestionnaire de files d'attente de l'agent est `SRCAGENTQM`, que le nom du gestionnaire de files d'attente de coordination est `COORDQM` et que l'agent redistribuable s'exécute à partir du répertoire `/home/myuser/mqmft-redis`, le chemin d'accès complet de ce fichier de socket est: `/home/myuser/mqmft-redis/mqft/logs/COORDQM/agents/SRCAGENT/logs/SRCAGENT@SRCAGENTQM`

qui correspond à une longueur totale de chemin de fichier de 85 caractères.

La longueur de chemin maximale autorisée par ces systèmes d'exploitation pour un fichier socket est de 107 caractères. Par conséquent, lors de la création d'un agent, veillez à ce que le chemin du fichier de socket ne dépasse pas 107 caractères. Ceci est particulièrement important avec un agent redistribuable dans lequel le répertoire de journaux de l'agent peut se trouver dans un emplacement de répertoire arbitraire. Pour plus de détails sur la configuration du répertoire de configuration, voir la commande **fteCreateEnvironment**.

Si vous démarrez un agent ou que d'autres commandes qui se connectent à l'agent sont exécutées et que la longueur de votre chemin dépasse 107 caractères, vous recevez le message suivant:

BFGNV0159E: Echec de la tentative de liaison au fichier de socket avec FFDC

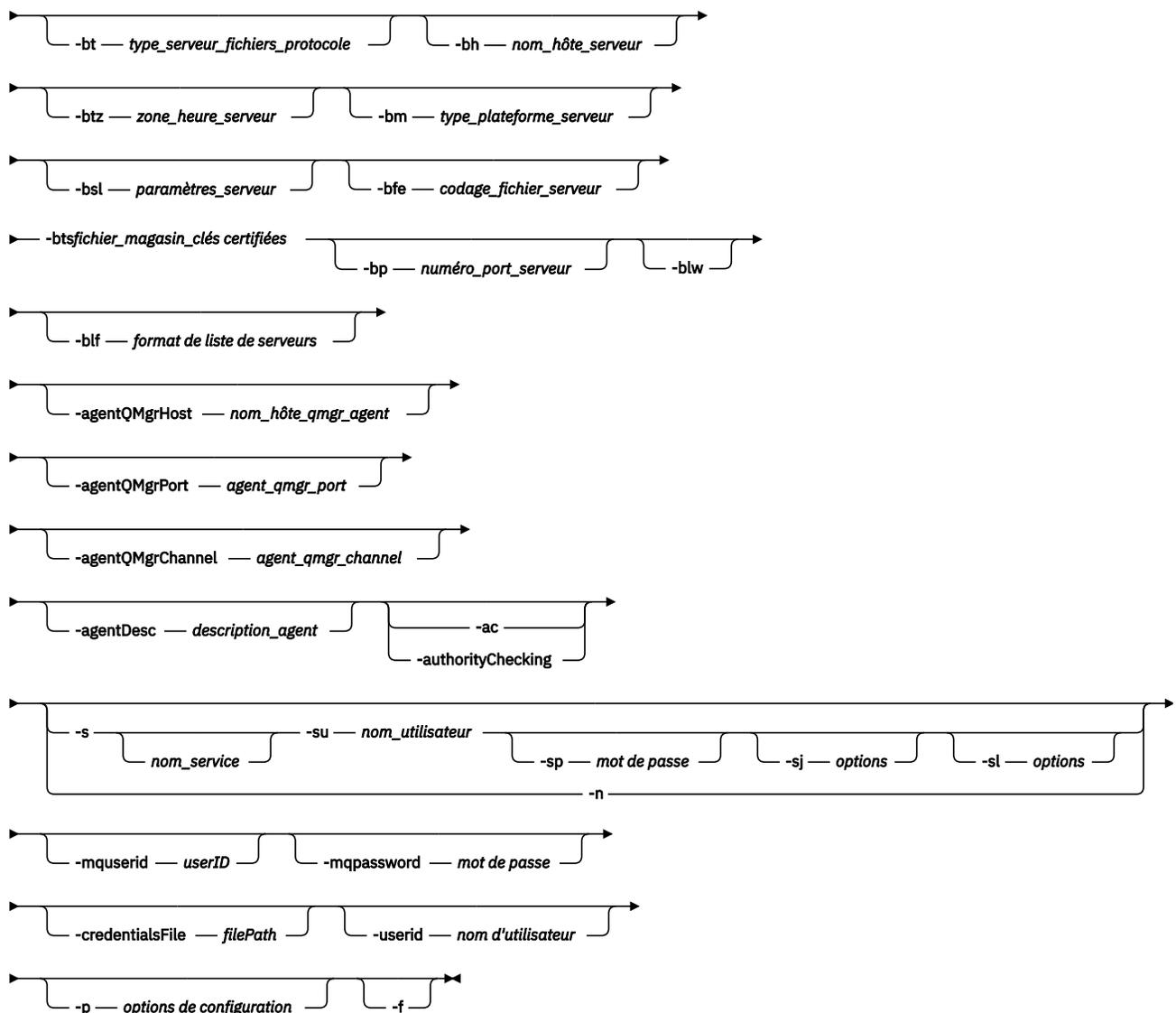
Caractères spéciaux

Soyez attentif lorsque vous utilisez des valeurs de paramètre contenant des caractères spéciaux pour que l'interpréteur de commandes interprète les caractères de façon appropriée. Par exemple, les noms et les chemins de fichier qualifiés complets contenant des caractères tels qu'un espace, une apostrophe ou des guillemets, ou bien une barre oblique ou une barre oblique inversée peuvent être interprétés par l'interpréteur de commandes au lieu d'être transmis directement à la commande. Pour éviter que des caractères soient interprétés par l'interpréteur de commandes, placez l'intégralité du paramètre entre apostrophes ou guillemets ou associez les caractères spéciaux à la séquence d'échappement de l'interpréteur de commandes.

Syntaxe

fteCreateBridgeAgent

► fteCreateBridgeAgent — -agentName *nom_agent* -agentQMgr *nom_gestionnaire_files_attente* →



Paramètres

-agentName (nom_agent)

Obligatoire. Nom de l'agent que vous souhaitez créer. Le nom de l'agent doit être unique dans son domaine d'administration.

Pour plus d'informations sur les agents de dénomination, voir [Conventions de dénomination d'objet](#).

-agentQMgr (agent_qmgr_name)

Obligatoire. Nom du gestionnaire de files d'attente d'agent.

-bt (type_serveur_fichier_protocole)

Facultatif. Indique que vous souhaitez définir un serveur de fichiers de protocole par défaut. Indiquez l'une des options suivantes :

protocole FTP

Serveur FTP standard

Protocole SFTP

Serveur FTP SSH

FTPS

Serveur FTP sécurisé à l'aide de SSL ou TLS

Si vous ne spécifiez pas ce paramètre, aucun serveur de protocole par défaut n'est défini.

-bh (nom_hôte_serveur)

Obligatoire uniquement si vous spécifiez également un serveur de fichiers de protocole par défaut à l'aide du paramètre **-bt** . Nom d'hôte IP ou adresse IP du serveur de fichiers de protocole.

-btz (fuseaux horaires du serveur)

Obligatoire uniquement si vous spécifiez également le paramètre **-bt** (serveurs FTP et FTPS uniquement). Fuseau horaire du serveur de fichiers de protocole. Indiquez le fuseau horaire au format suivant: Zone / Emplacement. Par exemple: Europe / Londres.

Vous pouvez utiliser le paramètre **-htz** pour répertorier les valeurs possibles pour **-btz**. Par exemple :`fteCreateBridgeAgent -htz`

-bm (plateau_serveur)

Obligatoire uniquement si vous spécifiez également un serveur de fichiers de protocole par défaut à l'aide du paramètre **-bt** . Type de plateforme du serveur de fichiers de protocole. Indiquez l'une des options suivantes :

UNIX

Plateformes UNIX et Linux génériques

WINDOWS

Plateforme Windows générique

-bsl (paramètres_serveur)

Obligatoire uniquement si vous spécifiez également le paramètre **-bt** (serveurs FTP et FTPS uniquement). Environnement local du serveur de fichiers de protocole. Indiquez l'environnement local au format suivant: `xx_XX`. Par exemple: `en_GB`.

- `xx` est le code de langue ISO. Pour obtenir la liste des valeurs valides, voir [Codes pour la représentation des noms de langues](#) .
- `XX` est le code pays ISO. Pour obtenir la liste des valeurs valides, voir [Noms de pays et éléments de code](#)

-bfe (encodage_fichier_serveur)

Obligatoire uniquement si vous spécifiez également un serveur de fichiers de protocole par défaut à l'aide du paramètre **-bt** . Format de codage de caractères des fichiers stockés sur le serveur de fichiers de protocole. Par exemple: UTF-8.

Vous pouvez utiliser le paramètre **-hcs** pour répertorier les valeurs possibles pour **-bfe**. Par exemple :`fteCreateBridgeAgent -hcs`

-bts (fichier_magasin_clés certifiées)

Obligatoire lorsque vous spécifiez le paramètre **-bt** (serveurs FTPS uniquement). Indique le chemin d'accès à un magasin de clés de confiance utilisé pour valider le certificat présenté par le serveur FTPS.

Vous pouvez spécifier le paramètre **-bts** uniquement si vous avez également spécifié l'option FTPS sur le paramètre **-bt**.

-bp (port_serveur)

Facultatif. Port IP auquel le serveur de fichiers de protocole est connecté. Spécifiez ce paramètre uniquement si votre serveur de fichiers de protocole n'utilise pas le port par défaut pour ce protocole. Si vous ne spécifiez pas ce paramètre, Managed File Transfer utilise le port par défaut pour le type de protocole du serveur de fichiers.

-blw

Facultatif. Définit le serveur de fichiers de protocole comme ayant des capacités d'écriture limitées. Par défaut, un agent de pont de protocole s'attend à ce que le serveur de fichiers de protocole autorise la suppression de fichier, le changement de nom de fichier et l'ouverture de fichier pour l'écriture d'ajout. Indiquez ce paramètre pour indiquer que le serveur de fichiers de protocole n'autorise pas ces actions sur les fichiers. A la place, le serveur de fichiers autorise uniquement la lecture et l'écriture dans le fichier. Si vous spécifiez ce paramètre, les transferts risquent de ne pas être récupérables s'ils sont interrompus et peuvent entraîner un échec pour le fichier en cours de transfert.

-blf (format de liste de serveurs)

Facultatif et pour les serveurs FTP et FTPS uniquement. Définit le format de liste de serveur des informations de fichier répertoriées renvoyées par le serveur de fichiers de protocole par défaut. Les options sont les suivantes :

UNIX

Plateformes UNIX et Linux génériques

WINDOWS

Plateforme Windows générique

Pour identifier le format à sélectionner, utilisez un programme client FTP et effectuez une liste d'un répertoire, puis sélectionnez le format qui convient le mieux. Par exemple :

 UNIX affiche le type de liste suivant:

```
-rwxr-xr-x 2 userid groupId 4096 2009-07-23 09:36 filename
```

 Windows affiche le type de liste suivant:

```
437,909 filename
```

 IBM i affiche le type de liste suivant:

```
0S400IFS -rwxrwsrwx 3 USERID 0 8192 Mar 7 08:33 filename
```

La valeur par défaut est UNIX, qui est le format utilisé par la plupart des serveurs.

-agentQMgrHôte (agent_qmgr_host)

Facultatif. Nom d'hôte ou adresse IP du gestionnaire de files d'attente de l'agent.

-agentQMGrPort (*agent_qmgr_port*)

Facultatif. Numéro de port utilisé pour les connexions client au gestionnaire de files d'attente d'agent.

-agentQMGrCanal (*agent_qmgr_channel*)

Facultatif. Nom de canal utilisé pour la connexion au gestionnaire de files d'attente d'agent.

-agentDesc (*description_agent*)

Facultatif. Description de l'agent, qui s'affiche dans IBM MQ Explorer.

-ac ou -authorityChecking

Facultatif. Ce paramètre permet le contrôle des droits d'accès.\n Si vous spécifiez ce paramètre, l'agent vérifie que les utilisateurs qui soumettent des demandes sont autorisés à effectuer l'action demandée. Pour plus d'informations, voir [Restriction des droits utilisateur sur les actions de l'agent MFT](#).

Windows -s (*nom_service*)

Facultatif (Windows uniquement). Indique que l'agent doit être exécuté en tant que service Windows . Si vous ne spécifiez pas *service_name*, le service est nommé mqmftAgentAGENTQMGR, où *AGENT* est le nom de l'agent et *QMGR* est le nom de votre gestionnaire de files d'attente d'agent.

Le nom d'affichage du service, qui est affiché dans la fenêtre Windows **Services** de la colonne **Nom**, est toujours **Managed File Transfer Agent AGENT@QMGR**.

Windows -su (*nom_utilisateur*)

Facultatif (Windows uniquement). Lorsque l'agent doit s'exécuter en tant que service Windows , ce paramètre indique le nom du compte sous lequel le service s'exécute. Pour exécuter l'agent à l'aide d'un compte utilisateur de domaine Windows , indiquez la valeur dans le formulaire DomainName\UserName. Pour exécuter le service à l'aide d'un compte à partir du domaine intégré local, spécifiez la valeur dans le formulaire UserName.

Le compte utilisateur Windows que vous spécifiez à l'aide du paramètre **-su** doit disposer du droit **Log on as a service** . Pour plus d'informations sur l'octroi de ce droit, voir [Conseils pour l'exécution d'un agent ou d'un consignateur MFT en tant que service Windows](#).

Obligatoire lorsque **-s** est spécifié.

Windows -sp (*mot_de_passe*)

Facultatif (Windows uniquement). Mot de passe du compte utilisateur défini par le paramètre **-su** .

Ce paramètre est valide uniquement lorsque **-s** est spécifié. Si vous ne spécifiez pas ce paramètre lorsque vous spécifiez le paramètre **-s** , un message d'avertissement est généré. Ce message vous avertit que vous devez définir le mot de passe à l'aide de l'outil Windows Services avant que le service ne démarre correctement.

Windows -sj (*options*)

Facultatif (Windows uniquement). Lorsque l'agent est démarré en tant que service Windows , définit une liste d'options sous la forme -D ou -X qui sont transmises à la machine virtuelle Java. Les options sont séparées par un signe dièse (#) ou un point-virgule (;). Si vous devez incorporer des caractères # ou point-virgule (;), placez-les entre apostrophes.

Ce paramètre est valide uniquement lorsque **-s** est spécifié.

Windows -sl (*options*)

Facultatif (Windows uniquement). Définit le niveau de journalisation du service Windows . Les options valides sont: error, info, warn, debug. La valeur par défaut est info. Cette option peut être utile si vous rencontrez des problèmes avec le service Windows . Le fait de le définir sur debug fournit des informations plus détaillées dans le fichier journal du service.

Ce paramètre est valide uniquement lorsque **-s** est spécifié.

Windows -n

Facultatif (Windows uniquement). Indique que l'agent doit être exécuté en tant que processus normal. Cette option est mutuellement exclusive avec l'option **-s**. Si aucun des paramètres **-s** et **-n** n'est spécifié, l'agent est configuré en tant que processus Windows normal.

-p (options-configuration)

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour créer un agent. Par convention, utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande **fteCreateBridgeAgent** utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Spécifiez le paramètre facultatif **-p** uniquement si vous souhaitez utiliser des options de configuration différentes de vos valeurs par défaut. Si vous ne spécifiez pas **-p**, les options de configuration définies dans le fichier `installation.properties` sont utilisées. Pour plus d'informations, voir [Options de configuration](#).

-f

Facultatif. Oblige la commande à écraser la configuration existante.

-htz

Facultatif. Affiche la liste des fuseaux horaires pris en charge que vous pouvez utiliser comme entrée pour le paramètre **-btz**.

-hcs

Facultatif. Affiche la liste des jeux de caractères pris en charge que vous pouvez utiliser comme entrée pour le paramètre **-bfe**.

Exécutez la commande **fteCreateBridgeAgent -hcs** pour répertorier les pages de codes connues pour la machine virtuelle Java. Ces informations ne sont pas disponibles à partir d'une source externe car les pages de codes connues varient d'une machine JVM à l'autre.

-mquserid (userID)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de files d'attente de commandes.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de commandes. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

-credentialsFile (filePath)

Facultatif. Chemin d'accès complet à un fichier de données d'identification existant ou nouveau, dans lequel les détails de l'authentification IBM MQ sont ajoutés.

Cette commande prend en charge l'ajout d'un ensemble de détails d'authentification IBM MQ à un fichier de données d'identification Managed File Transfer nommé. Utilisez-la si l'authentification de connexion IBM MQ a été activée. Si vous mettez à jour les détails existants, vous devez utiliser le paramètre force **-f**.

-userid (nom_utilisateur)

Facultatif. ID utilisateur indiqué pour associer les données d'identification détaillées. Si vous ne spécifiez pas d'ID utilisateur, les données d'identification détaillées sont valables pour tous les utilisateurs. Vous devez également spécifier le paramètre **-credentialsFile**.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Paramètres obsolètes

Les paramètres suivants sont obsolètes et ne sont pas pris en charge sous IBM WebSphere MQ 7.5 ou sous IBM WebSphere MQ File Transfer Edition 7.0.2 ou version ultérieure.

-brd (retard_reconnexion)

Obsolète. Facultatif. Indique, en secondes, le délai entre les tentatives de rétablissement d'une connexion perdue avec le serveur de fichiers de protocole. La valeur par défaut est de 10 secondes.

-brr (relances de connexion)

Obsolète. Facultatif. Indique le nombre maximal de nouvelles tentatives lors de la tentative de rétablissement d'une connexion perdue avec le serveur de fichiers de protocole par défaut. Lorsque ce nombre maximal est atteint, le transfert de fichier en cours est classé comme ayant échoué. La valeur par défaut est 2.

Exemples

Dans cet exemple, un nouvel agent de pont de protocole ACCOUNTS1 est créé avec un gestionnaire de files d'attente d'agent QM_ACCOUNTS et utilise le gestionnaire de files d'attente de coordination par défaut. ACCOUNTS1 se connecte au serveur FTP accountshost.ibm.com. Ce serveur FTP s'exécute sur Windows à l'aide du fuseau horaire Europe / Berlin, de l'environnement local de_DE et du codage de fichier UTF-8. Le nombre de tentatives de reconnexion est de 4:

```
fteCreateBridgeAgent -agentName ACCOUNTS1 -agentQMGr QM_ACCOUNTS -bt FTP
-bh accountshost.ibm.com -bm WINDOWS -btz Europe/Berlin -bsl de_DE -bfe UTF8
-agentQMGrHost myhost.ibm.com -agentQMGrPort 1415 -agentQMGrChannel CHANNEL1
```

Dans cet exemple, un nouvel agent de pont de protocole ACCOUNTS2 est créé avec un gestionnaire de files d'attente d'agent QM_ACCOUNTS et utilise le gestionnaire de coordination par défaut. ACCOUNTS2 est créé sans serveur de fichiers de protocole par défaut.

```
fteCreateBridgeAgent -agentName ACCOUNTS2 -agentQMGr QM_ACCOUNTS
```

Remarque : Ce qui précède ne s'applique pas à Managed File Transfer Agent redistribuable.

Personnalisation supplémentaire

Si vous avez utilisé le paramètre `-bt` (et les paramètres supplémentaires requis), le fichier `ProtocolBridgeProperties.xml` contient un nom de serveur par défaut.

Si vous souhaitez ajouter des serveurs ftp supplémentaires ou modifier l'emplacement du fichier de données d'identification, voir [Définition des propriétés des serveurs de fichiers de protocole à l'aide du fichier ProtocolBridgeProperties.xml](#).

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Utilisez la commande **fteStartAgent** pour démarrer votre agent de pont de protocole. Pour plus d'informations, voir la section «[fteStartAgent](#)», à la page 540.  Voir aussi [Démarrage d'un agent MFT sous z/OS](#).

Référence associée

«[Format de fichier des données d'identification du pont de protocole](#)», à la page 589

Le fichier `ProtocolBridgeCredentials.xml` du répertoire de configuration Managed File Transfer Agent définit les noms d'utilisateur et les données d'identification que l'agent de pont de protocole utilise pour s'autoriser avec le serveur de protocole.

«Format du fichier de propriétés du pont de protocole», à la page 592

Le fichier `ProtocolBridgeProperties.xml` du répertoire de configuration de l'agent définit les propriétés des serveurs de fichiers de protocole.

Information associée

[Le pont de protocole](#)

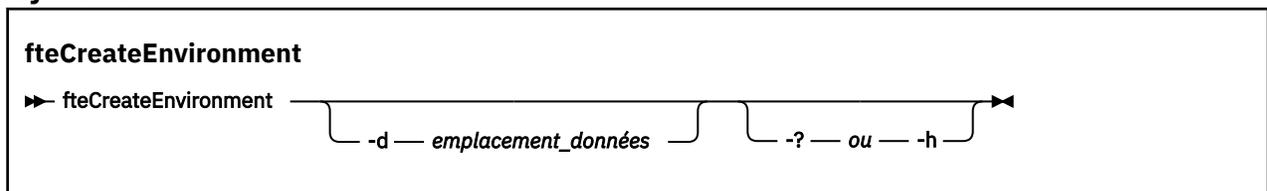
V 9.0.1 Environnement `fteCreate`(configuration de l'environnement pour Redistributable Managed File Transfer Agent)

La commande `fteCreateEnvironment` définit l'environnement pour la configuration et le transfert de fichiers pour Redistributable Managed File Transfer Agent.

Objet

La commande `fteCreateEnvironment` permet de configurer l'environnement pour l'utilisation du Redistributable Managed File Transfer Agent. Vous pouvez exécuter cette commande avec le paramètre `-d` pour spécifier l'emplacement des fichiers de données de l'agent MFT. Si vous ne spécifiez pas le paramètre `-d`, la commande crée les fichiers de données dans l'emplacement de téléchargement Redistributable Managed File Transfer Agent et définit le chemin de données.

Syntaxe



Paramètres

`-d (emplacement_données)`

Facultatif. Ce paramètre permet de spécifier l'emplacement des fichiers de données au moment de la configuration de l'environnement.

Si vous ne spécifiez pas ce paramètre, le répertoire de données (s'il n'existe pas déjà) est créé à l'emplacement où Redistributable Managed File Transfer Agent est extrait et la variable d'environnement (`BFG_DATA`) est définie pour cet emplacement.

`-? ou -h`

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, sous Windows, le paramètre `-d` indique l'emplacement où les dossiers de données sont créés:

```
fteCreateEnvironment -d C:\mftRedistributable\mftData
```

Sous Linux, la commande doit être exécutée sur un interpréteur de commandes bash. Dans un interpréteur de commandes bash, la commande peut être exécutée de différentes manières et le fichier de commandes doit être sourcé:

```
source Path_of_MFTZipBin/fteCreateEnvironment
```

Une autre méthode consiste à:

```
. Path_of_MFTZipBin/fteCreateEnvironment
```

ou, en cas d'exécution à partir du répertoire dans lequel se trouve le fichier de commandes:

```
./fteCreateEnvironment
```



Attention : Notez l'espace qui suit le premier point (.)

Codes retour

0

La commande a abouti.

1

La commande a échoué.

fteCreateCDAgent (créer un agent de pont Connect:Direct)

La commande fteCreateCDAgent crée un Managed File Transfer Agent et sa configuration associée à utiliser avec le pont Connect:Direct .

Important :  Sous IBM MQ for UNIX, Linux, and Windows, seuls les utilisateurs qui sont des administrateurs d'IBM MQ (et membres du groupe mqm) peuvent exécuter cette commande. Si vous tentez d'exécuter cette commande en tant qu'utilisateur qui n'est pas un administrateur IBM MQ , le message d'erreur BFGCL0502E: Vous n'êtes pas autorisé à effectuer l'opération demandée . et la commande n'est pas exécutée.

 Sur les systèmes z/OS, l'utilisateur doit remplir (au moins) l'une des conditions suivantes pour pouvoir exécuter la commande migrate :

- Etre membre du groupe mqm (si le groupe mqm est défini sur le système).
-  Etre membre du groupe désigné dans la variable d'environnement BFG_GROUP_NAME (si un tel groupe a été désigné).
-  Ne pas avoir de valeur définie dans la variable d'environnement BFG_GROUP_NAME lorsque la commande est exécutée.

Objet

La commande **fteCreateCDAgent** permet de créer un agent de pont Connect:Direct . Ce type d'agent est dédié au transfert de fichiers vers et depuis des noeuds Connect:Direct . Pour plus d'informations, voir [The Connect:Direct bridge](#). Pour plus d'informations sur les versions de système d'exploitation prises en charge pour le pont Connect:Direct , voir la page Web [System Requirements for IBM MQ](#).

Cette commande fournit les commandes MQSC que vous devez exécuter sur votre gestionnaire de files d'attente d'agent pour créer les files d'attente d'agent suivantes:

- 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*
- 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*

Ces files d'attente sont des files d'attente système internes à partir de laquelle vous ne devez pas modifier, supprimer ou lire des messages, sauf si vous supprimez l'agent. Les commandes MQSC à exécuter sont également fournies dans un fichier à l'emplacement suivant:

MQ_DATA_PATH\mqft\config\coordination_qmgr_name\agents\agent_name\agent_name_create.mqsc.

Si vous souhaitez ultérieurement supprimer l'agent, cette commande vous fournit également les commandes MQSC que vous devez exécuter pour effacer puis supprimer les files d'attente appartenant à l'agent. Les commandes MQSC se trouvent dans un fichier à l'emplacement suivant:

MQ_DATA_PATH\mqft\config\coordination_qmgr_name\agents\agent_name\agent_name_delete.mqsc.

Managed File Transfer fournit des propriétés d'agent avancées qui vous aident à configurer des agents. Ces propriétés sont décrites dans le [fichier MFT agent.properties](#).

La commande **fteCreateCDAgent** crée deux fichiers XML dans le répertoire des propriétés de l'agent. `ConnectDirectNodeProperties.xml`, qui permet de définir des informations sur les noeuds distants dans un transfert, et `ConnectDirectProcessDefinitions.xml`, qui permet de spécifier les processus Connect:Direct définis par l'utilisateur qui sont démarrés par des transferts.

Pour définir les noms d'utilisateur et les mots de passe utilisés par l'agent de pont Connect:Direct pour se connecter aux noeuds Connect:Direct, vous devez créer manuellement un fichier `ConnectDirectCredentials.xml`. Les exemples de fichiers XML se trouvent dans *MQ_INSTALLATION_PATH*/mqft/samples/credentials/. Pour plus d'informations et d'exemples, consultez «Format de fichier de données d'identification Connect:Direct», à la page 599.

Important :

Sur les plateformes UNIX et les commandes Linux Managed File Transfer, utilisez des fichiers de socket pour communiquer avec le processus d'agent exécuté sur la même machine hôte.

Ces fichiers de socket sont créés dans le répertoire de journaux de l'agent et sont supprimés lorsqu'un agent s'arrête. Dans l'installation de IBM MQ Managed File Transfer, ce fichier de socket est créé avec le chemin d'accès `<MQ_DATA_PATH>/mqft/logs/<COORDINATION_QM_NAME>/agents/<AGENT_NAME>/logs/<AGENT_NAME>@<AGENT_QM_NAME>` où `MQ_DATA_PATH` est `/var/mqm` par défaut.

Pour un agent redistribuable, ce fichier de socket est créé sous le répertoire: `<RE_DISTRIBUTABLE_DIRECTORY>/mqft/logs/<COORDINATION_QM_NAME>/agents/<AGENT_NAME>/logs/<AGENT_NAME>@<AGENT_QM_NAME>`.

Par exemple, si le nom de l'agent est SRCAGENT, que le nom du gestionnaire de files d'attente de l'agent est SRCAGENTQM, que le nom du gestionnaire de files d'attente de coordination est COORDQM et que l'agent redistribuable s'exécute à partir du répertoire `/home/myuser/mqmft-redis`, le chemin d'accès complet de ce fichier de socket est: `/home/myuser/mqmft-redis/mqft/logs/COORDQM/agents/SRCAGENT/logs/SRCAGENT@SRCAGENTQM`

qui correspond à une longueur totale de chemin de fichier de 85 caractères.

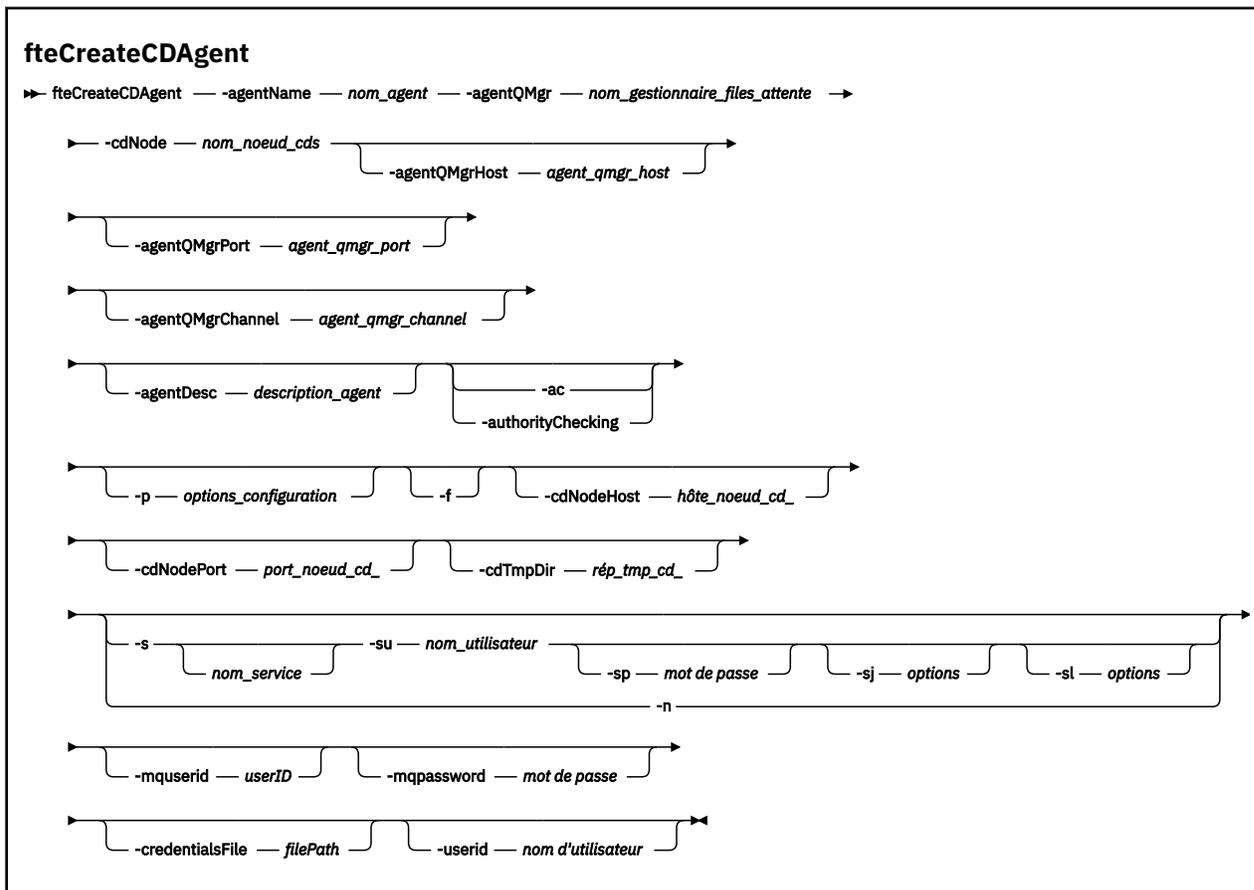
La longueur de chemin maximale autorisée par ces systèmes d'exploitation pour un fichier socket est de 107 caractères. Par conséquent, lors de la création d'un agent, veillez à ce que le chemin du fichier de socket ne dépasse pas 107 caractères. Ceci est particulièrement important avec un agent redistribuable dans lequel le répertoire de journaux de l'agent peut se trouver dans un emplacement de répertoire arbitraire. Pour plus de détails sur la configuration du répertoire de configuration, voir la commande **fteCreateEnvironment**.

Si vous démarrez un agent ou que d'autres commandes qui se connectent à l'agent sont exécutées et que la longueur de votre chemin dépasse 107 caractères, vous recevez le message suivant:

BFGNV0159E: Echec de la tentative de liaison au fichier de socket avec FFDC

Caractères spéciaux

Soyez attentif lorsque vous utilisez des valeurs de paramètre contenant des caractères spéciaux pour que l'interpréteur de commandes interprète les caractères de façon appropriée. Par exemple, les noms et les chemins de fichier qualifiés complets contenant des caractères tels qu'un espace, une apostrophe ou des guillemets, ou bien une barre oblique ou une barre oblique inversée peuvent être interprétés par l'interpréteur de commandes au lieu d'être transmis directement à la commande. Pour éviter que des caractères soient interprétés par l'interpréteur de commandes, placez l'intégralité du paramètre entre apostrophes ou guillemets ou associez les caractères spéciaux à la séquence d'échappement de l'interpréteur de commandes.



Paramètres

-agentName (*nom_agent*)

Obligatoire. Nom de l'agent que vous souhaitez créer. Le nom de l'agent doit être unique dans son gestionnaire de files d'attente de coordination.

Pour plus d'informations sur les agents de dénomination, voir [Conventions de dénomination d'objet](#).

-agentQMgr (*agent_qmgr_name*)

Obligatoire. Nom du gestionnaire de files d'attente d'agent.

-cdNode *nom_noeud_cd*

Obligatoire. Nom du noeud Connect:Direct à utiliser pour transférer les messages de cet agent vers les noeuds Connect:Direct de destination. La valeur de ce paramètre est utilisée pour la consignation et non pour indiquer à l'agent de pont Connect:Direct à quel noeud se connecter. Les valeurs de **-cdNodeHost** et **-cdNodePort** spécifient le noeud Connect:Direct qui fait partie du pont Connect:Direct.

-agentQMgrHôte (*agent_qmgr_host*)

Facultatif. Nom d'hôte ou adresse IP du gestionnaire de files d'attente de l'agent.

-agentQMgrPort (*agent_qmgr_port*)

Facultatif. Numéro de port utilisé pour les connexions client au gestionnaire de files d'attente d'agent.

-agentQMgrCanal (*agent_qmgr_channel*)

Facultatif. Nom de canal utilisé pour la connexion au gestionnaire de files d'attente d'agent.

-agentDesc (*description_agent*)

Facultatif. Description de l'agent, qui s'affiche dans IBM MQ Explorer.

-ac ou -authorityChecking

Facultatif. Ce paramètre permet le contrôle des droits d'accès.\n Si vous spécifiez ce paramètre, l'agent vérifie que les utilisateurs qui soumettent des demandes sont autorisés à effectuer l'action demandée. Pour plus d'informations, voir Restriction des droits utilisateur sur les actions de l'agent MFT.

-p (*options_configuration*)

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour créer un agent. Par convention, utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande **fteCreateCDAgent** utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Spécifiez le paramètre facultatif **-p** uniquement si vous souhaitez utiliser des options de configuration différentes de vos valeurs par défaut. Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

-f

Facultatif. Force la commande à remplacer les paramètres existants qui ne correspondent pas. La spécification de ce paramètre ne force pas le remplacement d'un agent de service Windows existant.

-cdNodeHôte *nom_hôte_noeud_cd*

Facultatif. Nom d'hôte ou adresse IP du système sur lequel se trouve le noeud Connect:Direct , spécifié par le paramètre **-cdNode** . Si vous ne spécifiez pas le paramètre **-cdNodeHost** , une valeur par défaut du nom d'hôte ou de l'adresse IP du système local est utilisée.

Dans la plupart des cas, le noeud Connect:Direct se trouve sur le même système que l'agent de pont Connect:Direct . Dans ces cas là, la valeur par défaut de cette propriété (l'adresse IP du système local) est correcte. Si votre système possède plusieurs adresses IP ou si votre noeud Connect:Direct se trouve sur un système différent de celui de votre agent de pont Connect:Direct et que leurs systèmes partagent un système de fichiers, utilisez cette propriété pour spécifier le nom d'hôte correct pour le noeud Connect:Direct .

-cdNodePort *nom_port_noeud_cd*

Facultatif. Numéro de port du noeud Connect:Direct utilisé par les applications client pour communiquer avec le noeud spécifié par le paramètre **-cdNode** . Dans la documentation du produit Connect:Direct, ce port est appelé port d'API. Si vous ne spécifiez pas le paramètre **-cdNodePort** , le numéro de port par défaut 1363 est utilisé.

-cdTmpDir *cd_tmp_directory*

Facultatif. Répertoire à utiliser par cet agent pour stocker temporairement les fichiers avant leur transfert vers le noeud Connect:Direct de destination. Ce paramètre indique le chemin d'accès complet du répertoire dans lequel les fichiers sont stockés temporairement. Par exemple, si **cdTmpDir** est défini sur /tmp , les fichiers sont placés temporairement dans le répertoire /tmp . Si vous ne spécifiez pas le paramètre **-cdTmpDir** , les fichiers sont stockés temporairement dans un répertoire nommé *cdbridge-agent_name*. Ce répertoire par défaut est créé à l'emplacement défini par la valeur de la propriété *java.io.tmpdir* .

Remarque : Si vous exécutez la commande **fteCleanAgent** , tous les fichiers de ce répertoire sont supprimés.

Windows **-s (nom_service)**

Facultatif (Windows uniquement). Indique que l'agent doit être exécuté en tant que service Windows , la commande doit être exécutée à partir d'un ID administrateur Windows . Si vous ne spécifiez pas *service_name*, le service est nommé `mqmftAgentAGENTQMGR`, où *AGENT* est le nom de l'agent et *QMGR* est le nom de votre gestionnaire de files d'attente d'agent.

Le nom d'affichage du service, qui est affiché dans la fenêtre Windows **Services** de la colonne **Nom** , est toujours **Managed File Transfer Agent AGENT@QMGR**.

Remarque : Si l'agent redistribuable doit s'exécuter en tant que service Windows, vous devez définir la variable d'environnement **BFG_DATA** dans l'environnement système pour que le service fonctionne.

Windows **-su (nom_utilisateur)**

Facultatif (Windows uniquement). Lorsque l'agent doit s'exécuter en tant que service Windows , ce paramètre indique le nom du compte sous lequel le service s'exécute. Pour exécuter l'agent à l'aide d'un compte utilisateur de domaine Windows , indiquez la valeur dans le formulaire `DomainName\UserName`. Pour exécuter le service à l'aide d'un compte à partir du domaine intégré local, spécifiez la valeur dans le formulaire `UserName`.

Le compte utilisateur Windows que vous spécifiez à l'aide du paramètre **-su** doit disposer du droit **Log on as a service** . Pour plus d'informations sur l'octroi de ce droit, voir [Conseils pour l'exécution d'un agent ou d'un consignateur MFT en tant que service Windows](#).

Obligatoire lorsque **-s** est spécifié.

Windows **-sp (mot_de_passe)**

Facultatif (Windows uniquement).

Ce paramètre est valide uniquement lorsque **-s** est spécifié. Si vous ne spécifiez pas ce paramètre lorsque vous spécifiez le paramètre **-s** , un message d'avertissement est généré. Ce message vous avertit que vous devez définir le mot de passe à l'aide de l'outil Windows Services avant que le service ne démarre correctement.

Windows **-sj (options)**

Facultatif (Windows uniquement). Lorsque l'agent est démarré en tant que service Windows , définit une liste d'options sous la forme `-D` ou `-X` qui sont transmises à la machine virtuelle Java. Les options sont séparées par un signe dièse (#) ou un point-virgule (;). Si vous devez incorporer des caractères # ou point-virgule (;), placez-les entre apostrophes.

Ce paramètre est valide uniquement lorsque **-s** est spécifié.

Windows **-sl (options)**

Facultatif (Windows uniquement). Définit le niveau de journalisation du service Windows . Les options valides sont: `error`, `info`, `warn`, `debug`. La valeur par défaut est `info`. Cette option peut être utile si vous rencontrez des problèmes avec le service Windows . Le fait de le définir sur `debug` fournit des informations plus détaillées dans le fichier journal du service.

Ce paramètre est valide uniquement lorsque **-s** est spécifié.

Windows **-n**

Facultatif (Windows uniquement). Indique que l'agent doit être exécuté en tant que processus normal. Cette option est mutuellement exclusive avec l'option **-s** . Si aucun des paramètres **-s** et **-n** n'est spécifié, l'agent est configuré en tant que processus Windows normal.

-mquserid (userID)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de files d'attente de commandes.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de commandes. Vous devez également spécifier le paramètre **-mquserid** . Si vous spécifiez

-mquserid, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

-credentialsFile (filePath)

Facultatif. Chemin d'accès complet à un fichier de données d'identification existant ou nouveau, dans lequel les détails de l'authentification IBM MQ sont ajoutés.

Cette commande prend en charge l'ajout d'un ensemble de détails d'authentification IBM MQ à un fichier de données d'identification Managed File Transfer nommé. Utilisez-la si l'authentification de connexion IBM MQ a été activée. Si vous mettez à jour les détails existants, vous devez utiliser le paramètre force **-f**.

-userid (nom_utilisateur)

Facultatif. ID utilisateur indiqué pour associer les données d'identification détaillées. Si vous ne spécifiez pas d'ID utilisateur, les données d'identification détaillées sont valables pour tous les utilisateurs. Vous devez également spécifier le paramètre **-credentialsFile**.

Exemple

Dans cet exemple, un nouvel agent de pont Connect:Direct CD_BRIDGE est créé avec un gestionnaire de files d'attente d'agent QM_NEPTUNE. L'agent utilise le noeud Connect:Direct BRIDGE_NODE pour transférer des fichiers vers d'autres noeuds Connect:Direct. Le noeud BRIDGE_NODE se trouve sur le même système que l'agent et utilise le port par défaut pour les connexions client. Les fichiers transférés vers ou depuis Connect:Direct sont temporairement stockés dans le répertoire /tmp/cd-bridge.

```
fteCreateCDAgent -agentName CD_BRIDGE -agentQMgr QM_NEPTUNE  
-cdNode BRIDGE_NODE -cdTmpDir /tmp/cd-bridge
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

fteCreateLogger (création d'un fichier MFT ou d'un consignateur de base de données)

Utilisez la commande **fteCreateLogger** pour créer un fichier Managed File Transfer ou un consignateur de base de données.

Important :  Sous IBM MQ for UNIX, Linux, and Windows, seuls les utilisateurs qui sont des administrateurs d'IBM MQ (et membres du groupe mqm) peuvent exécuter cette commande. Si vous tentez d'exécuter cette commande en tant qu'utilisateur qui n'est pas un administrateur IBM MQ, le message d'erreur BFGCL0502E: Vous n'êtes pas autorisé à effectuer l'opération demandée. et la commande n'est pas exécutée.

 Sur les systèmes z/OS, l'utilisateur doit remplir (au moins) l'une des conditions suivantes pour pouvoir exécuter la commande migrate :

- Etre membre du groupe mqm (si le groupe mqm est défini sur le système).
-  Etre membre du groupe désigné dans la variable d'environnement BFG_GROUP_NAME (si un tel groupe a été désigné).
-  Ne pas avoir de valeur définie dans la variable d'environnement BFG_GROUP_NAME lorsque la commande est exécutée.

Consignateurs sous IBM i

IBM i

Les programmes de consignation Managed File Transfer ne sont pas pris en charge sur la plateforme IBM i.

Objet

La commande **fteCreateLogger** fournit les commandes MQSC que vous devez exécuter sur votre gestionnaire de files d'attente de commandes de consignateur pour créer les files d'attente de consignateur suivantes:

- SYSTEME.SYSTEM.FTE.LOG.CMD.*nom_journal*
- SYSTEME.SYSTEM.FTE.LOG.RJCT.*nom_connexion*

Ces files d'attente sont des files d'attente internes du système dont vous ne devez pas modifier, supprimer ou lire les messages à moins que vous ne supprimiez le consignateur. Les commandes MQSC à exécuter sont également fournies dans un fichier à l'emplacement suivant:

`MQ_DATA_PATH\mqft\config\coordination_qmgr\loggers\logger_name\logger_name_create.mqsc`

Si vous souhaitez ultérieurement supprimer le consignateur, utilisez la commande **fteDeleteLogger**.

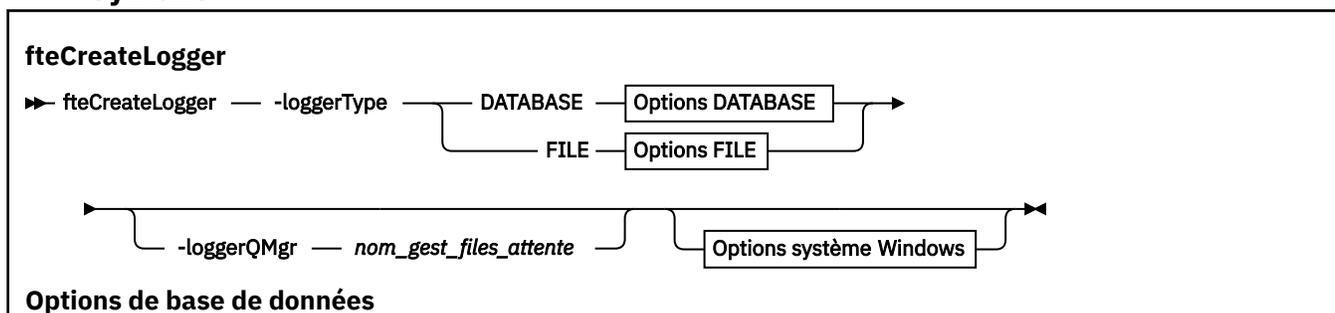
Managed File Transfer fournit des propriétés de consignateur avancées qui vous aident à configurer des consignateurs. Voir [Propriétés de configuration du consignateurMFT](#)

Remarque : Si le consignateur que vous créez est un consignateur de base de données et qu'il ne se connecte pas à une base de données Db2 locale, vous devez créer manuellement un fichier `MQMFTCredentials.xml`. Le fichier contient le nom d'utilisateur et le mot de passe pour la connexion à la base de données. Vous devez utiliser le fichier de propriétés, `wmqfte.database.credentials`, dans le fichier `logger.properties` pour spécifier le chemin d'accès au fichier `MQMFTCredentials.xml`. Un exemple de ce fichier de données d'identification se trouve dans `MQ_INSTALLATION_PATH/mqft/samples/credentials/`.

Caractères spéciaux

Soyez attentif lorsque vous utilisez des valeurs de paramètre contenant des caractères spéciaux pour que l'interpréteur de commandes interprète les caractères de façon appropriée. Par exemple, les noms et les chemins de fichier qualifiés complets contenant des caractères tels qu'un espace, une apostrophe ou des guillemets, ou bien une barre oblique ou une barre oblique inversée peuvent être interprétés par l'interpréteur de commandes au lieu d'être transmis directement à la commande. Pour éviter que des caractères soient interprétés par l'interpréteur de commandes, placez l'intégralité du paramètre entre apostrophes ou guillemets ou associez les caractères spéciaux à la séquence d'échappement de l'interpréteur de commandes.

Syntaxe



-dbName (nom_base_données)

Obligatoire lorsque -loggerType est DATABASE. Nom de la base de données dans laquelle sont stockées les informations de transfert de fichiers gérés. La base de données doit être configurée avec les tables de journal Managed File Transfer .

-dbDriver (pilote)

Obligatoire lorsque -loggerType est DATABASE. Emplacement des classes du pilote JDBC pour la base de données. Il s'agit généralement du chemin et du nom d'un fichier JAR.

-dbLib (chemin)

Facultatif lorsque -loggerType est DATABASE. Emplacement des bibliothèques natives requises par le pilote de base de données de votre choix.

-ModefileLogger(mode)

Obligatoire lorsque -loggerType est FILE. Indique le type de système de fichiers utilisé pour le stockage des informations de transfert de fichiers gérés. Les options sont LINEAR ou CIRCULAIRE.

L'option LINEAR signifie que le consignateur de fichier écrit les informations dans un fichier jusqu'à ce que ce fichier atteigne sa taille maximale définie par -filesize. Lorsque la taille maximale est atteinte, le consignateur de fichiers démarre un nouveau fichier. Les fichiers précédemment écrits ne seront pas supprimés, ce qui permet de les conserver en tant qu'enregistrement historique des messages de journal. Les fichiers ne sont pas supprimés lors de l'exécution dans ce mode. Par conséquent, -fileCount sera ignoré car il n'y a pas de limite supérieure au nombre de fichiers pouvant être créés. Comme il n'y a pas de limite supérieure lors de l'exécution dans ce mode, il sera nécessaire de suivre la quantité d'espace disque utilisée par les fichiers journaux afin d'éviter une faible quantité d'espace disque.

L'option CIRCULAIRE signifie que le consignateur de fichiers écrit les informations dans un fichier jusqu'à ce que ce fichier atteigne sa taille maximale définie par -fileSize. Lorsque la taille maximale est atteinte, le consignateur de fichiers démarre un nouveau fichier. Le nombre maximal de fichiers écrits dans ce mode est contrôlé par la valeur définie à l'aide de -fileCount. Lorsque ce nombre maximal de fichiers est atteint, le consignateur de fichiers supprime le premier fichier et le recrée pour l'utiliser comme fichier actif. Si la valeur définie dans -fileSize est une unité d'octet de taille fixe, la limite supérieure de l'espace disque utilisé dans ce mode sera égale à fileSize x fileCount. Si les valeurs définies dans -fileSize sont une unité de temps, la taille maximale dépend du débit des messages de journal dans votre système au cours de ces périodes.

Pour plus d'informations, voir [Propriétés de configuration du consignateurMFT](#)

-fileSize (taille)

Obligatoire lorsque -loggerType est FILE. Taille maximale qu'un fichier journal peut atteindre. La valeur est un entier positif, supérieur à zéro, suivi de l'une des unités suivantes: Ko, Mo, Go, m (minutes), h (heures), d (jours), w (semaines). Par exemple:-fileSize 5MB (indique une taille maximale de 5MB),-fileSize 2d (indique une valeur maximale de 2 jours de données).

-fileCount (nombre)

Obligatoire lorsque -loggerType est FILE et que -fileLoggerMode est CIRCULAIRE. Nombre maximal de fichiers journaux à créer. Lorsque la quantité de données dépasse la quantité maximale pouvant être stockée dans ce nombre de fichiers, le fichier le plus ancien est supprimé de sorte que le nombre de fichiers journaux ne dépasse jamais la valeur spécifiée dans ce paramètre.

-loggerQMgrHôte

Nom d'hôte ou adresse IP de la machine sur laquelle le gestionnaire de files d'attente du consignateur est en cours d'exécution.

La valeur par défaut est Aucun.

Si vous ne spécifiez pas le paramètre **-loggerQMgrHost** , le consignateur est créé en mode liaisons.

-PortloggerQMgr

Numéro de port sur lequel le gestionnaire de files d'attente du consignateur est à l'écoute.

La valeur par défaut est 1414.

-CanalloggerQMgr

Nom du canal utilisé pour la connexion au gestionnaire de files d'attente du consignateur.

La valeur par défaut est SYSTEM.DEF.SVRCONN.

Windows -s (nom_service)

Facultatif (systèmes Windows uniquement). Indique que le consignateur doit être exécuté en tant que service Windows . Si vous ne spécifiez pas *service_name*, le service est nommé `mqmftLoggerLOGGERQMGR`, où *LOGGER* est le nom du consignateur et *QMGR* est le nom de votre gestionnaire de files d'attente de consignateur.

Le nom d'affichage du service, qui est affiché dans la fenêtre **Services** Windows dans la colonne **Nom** , est toujours **Managed File Transfer Logger *LOGGER@QMGR***.

Windows -su (nom_utilisateur)

Facultatif (Windows uniquement). Lorsque le consignateur doit s'exécuter en tant que service Windows , ce paramètre indique le nom du compte sous lequel le service s'exécute. Pour exécuter le consignateur à l'aide d'un compte utilisateur de domaine Windows , spécifiez la valeur dans le formulaire `DomainName\UserName`. Pour exécuter le service à l'aide d'un compte à partir du domaine intégré local, spécifiez la valeur dans le formulaire `UserName`.

Le compte utilisateur Windows que vous spécifiez à l'aide du paramètre **-su** doit disposer du droit **Log on as a service** . Pour plus d'informations sur l'octroi de ce droit, voir [Conseils pour l'exécution d'un agent ou d'un consignateur MFT en tant que service Windows](#).

Obligatoire lorsque **-s** est spécifié.

Windows -sp (mot_de_passe)

Facultatif (Windows uniquement). Mot de passe du compte utilisateur défini par le paramètre **-su** .

Ce paramètre est valide uniquement lorsque **-s** est spécifié. Si vous ne spécifiez pas ce paramètre lorsque vous spécifiez le paramètre **-s** , un message d'avertissement est généré. Ce message vous avertit que vous devez définir le mot de passe à l'aide de l'outil Windows Services avant que le service ne démarre correctement.

Windows -sj (options)

Facultatif (Windows uniquement). Lorsque le consignateur est démarré en tant que service Windows , définit une liste d'options sous la forme `-D` ou `-X` qui sont transmises à la machine virtuelle Java. Les options sont séparées par un signe dièse (`#`) ou un point-virgule (`;`). Si vous devez imbriquer des caractères (`#`) ou point-virgule (`;`), placez-les entre apostrophes.

Ce paramètre est valide uniquement lorsque **-s** est spécifié.

Windows -sl (options)

Facultatif (Windows uniquement). Définit le niveau de journalisation du service Windows . Les options valides sont: `error`, `info`, `warn`, `debug`. La valeur par défaut est `info`. Cette option peut être utile si vous rencontrez des problèmes avec le service Windows . Le fait de le définir sur `debug` fournit des informations plus détaillées dans le fichier journal du service.

Ce paramètre est valide uniquement lorsque **-s** est spécifié.

-p (options de configuration)

Facultatif. Indique l'ensemble des options de configuration utilisées pour créer le consignateur. Par convention, cette valeur est le nom d'un gestionnaire de files d'attente de coordination. Si vous ne spécifiez pas ce paramètre, l'ensemble d'options de configuration par défaut est utilisé.

-f

Facultatif. Oblige la commande à écraser la configuration existante.

(nom_journal)

Obligatoire. Nom du signataire à créer. Ce nom est incorporé dans les noms de file d'attente Managed File Transfer et ne doit donc contenir que des lettres, des chiffres et des points (.) et des caractères de soulignement (_). Il est également limité à une longueur maximale de 28 caractères.

-credentialsFile (filePath)

Facultatif. Chemin d'accès complet à un fichier de données d'identification existant ou nouveau, dans lequel les détails de l'authentification IBM MQ sont ajoutés.

Cette commande prend en charge l'ajout d'un ensemble de détails d'authentification IBM MQ à un fichier de données d'identification Managed File Transfer nommé. Utilisez-la si l'authentification de connexion IBM MQ a été activée. Si vous mettez à jour les détails existants, vous devez utiliser le paramètre force **-f**.

-userid (nom_utilisateur)

Facultatif. ID utilisateur indiqué pour associer les données d'identification détaillées. Si vous ne spécifiez pas d'ID utilisateur, les données d'identification détaillées sont valables pour tous les utilisateurs. Vous devez aussi spécifier le paramètre **-credentialsFile**.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemples

Dans cet exemple, un signataire de fichier circulaire appelé filelogger1 est créé. Le signataire de fichiers crée un maximum de 10 fichiers, chaque fichier ayant une taille de 10MB, avec un maximum de 100MB d'espace disque au total:

```
fteCreateLogger -loggerType FILE -fileLoggerMode CIRCULAR -fileSize 10MB -fileCount 10 filelogger1
```

Dans cet exemple, un signataire de base de données appelé dblogger1 est créé. Le signataire de base de données se connecte à une base de données Db2 appelée FTEDB:

```
fteCreateLogger -loggerType DATABASE -dbName FTEDB -dbType DB2 -dbDriver "C:\Program Files (x86)\IBM\SQLLIB\java\db2jcc4.jar" dblogger1
```

Dans cet exemple, un signataire de base de données appelé dblogger1 est créé. Le signataire de base de données se connecte à une base de données Oracle appelée FTEDB:

```
fteCreateLogger -loggerType DATABASE -dbName FTEDB -dbType oracle -dbDriver "C:\app\oracle\product\12.1.0\dbhome_2\jdbc\lib\ojdbc7.jar" dblogger1
```

Dans cet exemple, un signataire de fichiers en mode client est créé à l'aide du nom d'hôte et du port et du canal par défaut:

```
fteCreateLogger -loggerType FILE -loggerQMGr CORDQM -loggerQMGrHost cordqm.ibm.com -fileLoggerMode CIRCULAR -fileSize 10MB -fileCount 10 FL1
```

Dans cet exemple, un signataire de fichiers en mode client est créé à l'aide du nom d'hôte, du port et du canal:

```
fteCreateLogger -loggerType FILE -loggerQMGr CORDQM -loggerQMGrHost cordqm.ibm.com -loggerQMGrPort 4444 -loggerQMGrChannel LOGGER_CHANNEL -fileLoggerMode CIRCULAR -fileSize 10MB -fileCount 10 FL1
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

fteCreateMonitor: créer un moniteur de ressources MFT

La commande **fteCreateMonitor** crée et démarre un nouveau moniteur de ressources à partir de la ligne de commande. Vous pouvez surveiller une ressource (par exemple, le contenu d'un répertoire) à l'aide de Managed File Transfer de sorte que lorsqu'une condition de déclenchement est remplie, une tâche spécifiée, telle qu'un transfert de fichier, est démarrée.

Objet

Utilisez la commande **fteCreateMonitor** pour créer, puis démarrer un nouveau moniteur de ressources à l'aide d'un agent Managed File Transfer . Par exemple, vous pouvez utiliser un moniteur de ressources de la manière suivante: une application externe place un ou plusieurs fichiers dans un répertoire connu et, lorsque le traitement est terminé, l'application externe place un fichier de déclenchement dans un répertoire contrôlé. Le fichier de déclenchement est ensuite détecté et un transfert de fichier défini démarre et copie les fichiers du répertoire connu vers un agent de destination.

Vous pouvez utiliser les paramètres **-ox** et **-ix** pour exporter et importer une configuration de moniteur de ressources dans un fichier XML. L'importation de ce fichier à l'aide de la commande **fteCreateMonitor** crée un nouveau moniteur de ressources avec les mêmes paramètres que le moniteur de ressources indiqué dans la commande **fteCreateMonitor** pour l'exportation vers le fichier XML. En outre, vous pouvez utiliser les paramètres **-f** et **-c** pour remplacer dynamiquement une configuration de moniteur.

Remarque : Il n'y a pas de restriction sur le nombre de moniteurs de ressources pouvant être créés sur un agent et tous exécutés avec la même priorité. Tenez compte des implications du chevauchement des ressources surveillées, des conditions de déclenchement conflictuelles et de la fréquence d'interrogation des ressources. Pour plus d'informations, voir [MFT Resource Monitoring Concepts](#) .

La commande **fteCreateMonitor** n'est pas prise en charge sur les agents de pont de protocole.

Conseil : Vous pouvez également utiliser la commande **fteListMonitors** pour exporter des configurations de moniteur de ressources dans un fichier XML:

- L'utilisation de la commande **fteListMonitors** avec **-ox** exporte la définition d'un moniteur de ressources unique.
-  **9.0.5** Depuis la IBM MQ 9.0.5, l'utilisation de la commande **fteListMonitor** avec **-od** exporte plusieurs définitions de moniteur de ressources vers un répertoire spécifié. Vous pouvez également utiliser l'option **-od** pour exporter une définition de moniteur de ressources unique vers un répertoire spécifié.

Pour plus d'informations sur la commande **fteListMonitors**, voir [«fteListMoniteurs: répertorie les moniteurs de ressources MFT»](#), à la page 487.

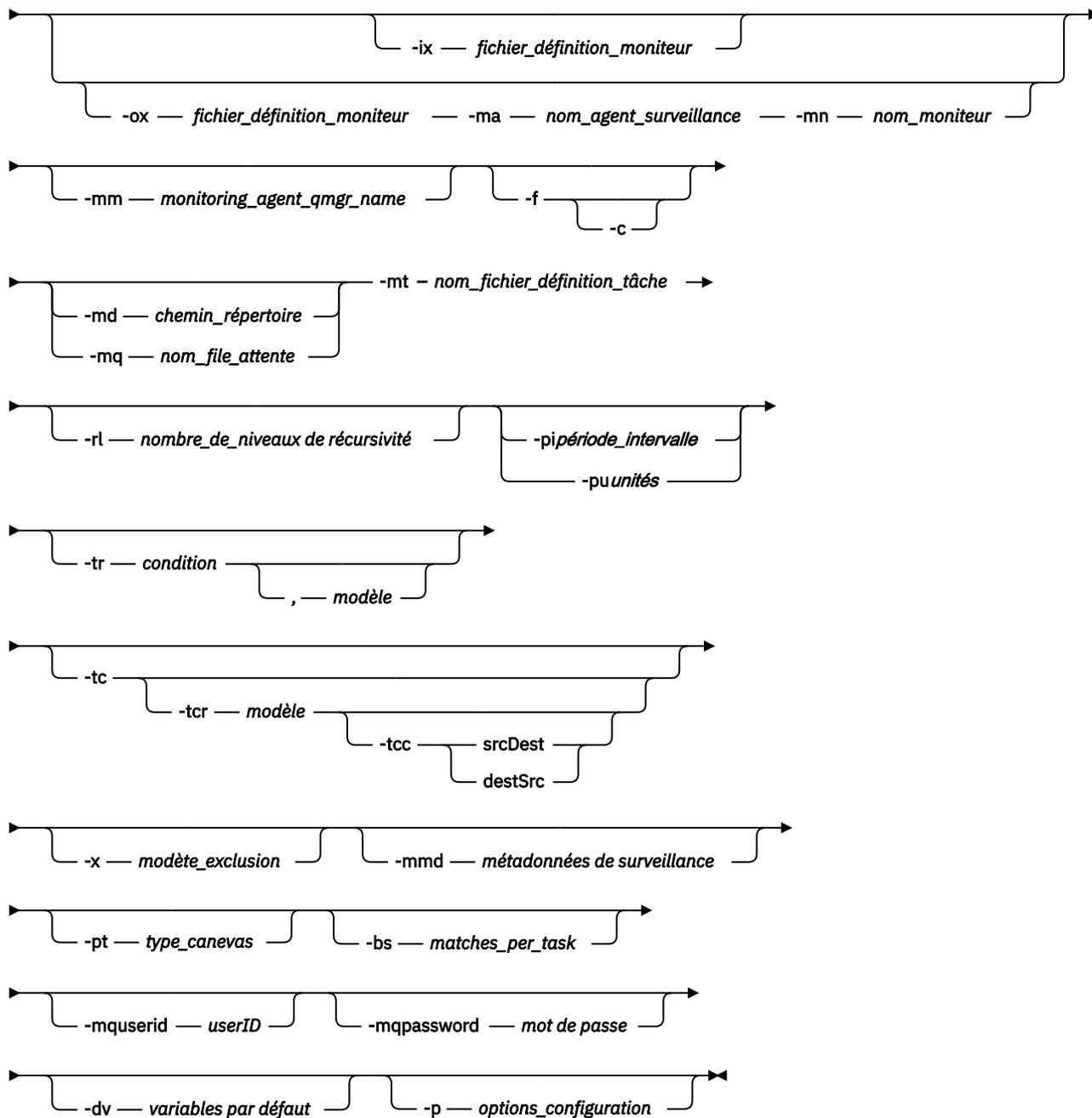
Caractères spéciaux

Soyez attentif lorsque vous utilisez des valeurs de paramètre contenant des caractères spéciaux pour que l'interpréteur de commandes interprète les caractères de façon appropriée. Par exemple, les noms et les chemins de fichier qualifiés complets contenant des caractères tels qu'un espace, une apostrophe ou des guillemets, ou bien une barre oblique ou une barre oblique inversée peuvent être interprétés par l'interpréteur de commandes au lieu d'être transmis directement à la commande. Pour éviter que des caractères soient interprétés par l'interpréteur de commandes, placez l'intégralité du paramètre entre apostrophes ou guillemets ou associez les caractères spéciaux à la séquence d'échappement de l'interpréteur de commandes.

Syntaxe

fteCreateMonitor

► fteCreateMonitor ►



Paramètres

-ix (*nom_fichier_xml*)

Facultatif. Importe la configuration du moniteur de ressources à partir d'un fichier XML.

-ox (*nom_fichier_xml*)

Facultatif. Ce paramètre doit être spécifié avec les paramètres `-ma` et `-mn`. Exporte la configuration du moniteur de ressources dans un fichier XML.

-mn (nom_moniteur)

Obligatoire. Nom que vous affectez à ce moniteur. Le nom du moniteur doit être unique pour l'agent de surveillance. Toutefois, vous pouvez supprimer un moniteur, puis créer un moniteur portant le même nom.

La longueur maximale d'un nom de moniteur de ressources est de 256 caractères. Les noms de moniteur de ressources ne sont pas sensibles à la casse. Les noms de moniteur de ressources entrés en minuscules ou en casse mixte sont convertis en majuscules. Les noms de moniteur de ressources ne doivent pas comporter d'astérisque (*), de signe pourcentage (%) ni de point d'interrogation (?).

-ma (nom_agent_surveillance)

Obligatoire. Nom de l'agent chargé de la surveillance des ressources. Cet agent de surveillance doit être l'agent source de la tâche de surveillance à déclencher.

-mm (monitoring_agent_qmgr_name)

Nom du gestionnaire de files d'attente auquel l'agent de surveillance est connecté. Etant donné que l'agent de surveillance et l'agent source doivent être identiques, ce gestionnaire de files d'attente est également votre gestionnaire de files d'attente d'agent source.

Remarque : La commande **fteCreateMonitor** se connecte au gestionnaire de files d'attente de commandes pour une topologie Managed File Transfer . Si le gestionnaire de files d'attente de commandes est également le gestionnaire de files d'attente de l'agent de surveillance, ce paramètre est facultatif. Sinon, le paramètre est obligatoire.

-f

Facultatif. Utilisez ce paramètre pour remplacer une configuration de moniteur de ressources. Par exemple, lorsque le nom du moniteur de ressources que vous choisissez existe déjà sur l'agent de surveillance des ressources et que vous souhaitez le mettre à jour au lieu de supprimer et de recréer un moniteur portant le même nom. L'utilisation de ce paramètre entraîne le redémarrage du processus de surveillance par l'agent.

-c

Facultatif. Ce paramètre efface l'historique d'un moniteur de ressources mis à jour, ce qui permet au moniteur de ressources de vérifier à nouveau les conditions de déclenchement. Vous pouvez utiliser ce paramètre uniquement avec le paramètre **-f** .

-md (chemin_répertoire)

Facultatif. Nom absolu du chemin de répertoire à surveiller. Sauf si vous utilisez les paramètres **-ix** ou **-ox** , vous devez spécifier l'un des paramètres **-md** ou **-mq** .

-mq (nom_file)

Facultatif. Nom de la file d'attente à surveiller. Cette file d'attente doit se trouver sur le gestionnaire de files d'attente de l'agent de surveillance. Sauf si vous utilisez les paramètres **-ix** ou **-ox** , vous devez spécifier l'un des paramètres **-md** ou **-mq** .

-mt (nom_fichier_définition_tâche)

Obligatoire. Nom du document XML qui contient la définition de tâche que vous souhaitez exécuter lorsque la condition de déclencheur est satisfaite. Pour plus d'informations, voir la section «[Utilisation de fichiers de définition de transfert](#)», à la page 164. Le chemin d'accès au document XML de définition de transfert doit se trouver sur le système de fichiers local à partir duquel vous exécutez la commande **fteCreateMonitor** . Si vous n'indiquez pas de chemin d'accès au fichier, la commande le recherche dans le répertoire de travail en cours. A moins que vous n'utilisiez les paramètres **-ix** ou **-ox** , **-mt** est un paramètre obligatoire.

Vous pouvez utiliser le paramètre **-gt** dans la commande de transfert **fteCreate** pour générer un modèle de document XML contenant votre demande de transfert de fichier. Le moniteur utilise le modèle de transfert comme définition de tâche.

V 9.0.1 Vous pouvez également utiliser le délai d'attente de reprise du transfert, le paramètre **-rt** , avec le paramètre **-gt** , lorsque vous exécutez la commande **fteCreateMonitor** . Vous pouvez définir la durée en secondes pendant laquelle l'agent source continue de se relier pour

recupérer un transfert qui est bloqué. Le paramètre de délai de récupération est ensuite inclus dans le document XML avec la définition de transfert utilisée par le moniteur. Pour plus d'informations sur la définition de ce paramètre, voir [Commande de transfertfiteCreate](#).

z/OS Sous z/OS, vous devez stocker le document de définition de tâche dans un fichier UNIX sur z/OS UNIX System Services. Vous ne pouvez pas stocker des documents de définition de tâche dans des fichiers séquentiels ou des membres PDS z/OS .

IBM i Sous IBM i, vous devez stocker le document de définition de tâche dans le système de fichiers intégré.

-rl (nombre_de_niveaux_récurtivité)

Facultatif. Niveau de récursivité de surveillance du répertoire de surveillance racine qui correspond au nombre de niveaux de sous-répertoire dans lesquels descendre. Par exemple, dans une structure de répertoire comme dans l'exemple suivant avec C:\wmqfte\monitor défini comme répertoire de surveillance racine

```
C:\wmqfte\monitor
C:\wmqfte\monitor\reports
C:\wmqfte\monitor\reports\2009
C:\wmqfte\monitor\reports\2009\April
```

Si vous spécifiez `-rl 2`, Managed File Transfer effectue des recherches uniquement dans le répertoire C:\wmqfte\monitor\reports\2009 et ses répertoires apparentés. Le répertoire C:\wmqfte\monitor\reports\2009\April est ignoré. Par défaut, la récursivité est définie sur none.

-pi (période_intervalle)

Facultatif. Intervalle entre chaque moniteur d'un répertoire. L'intervalle d'interrogation doit être une valeur entière positive. La valeur par défaut de `-pi` est 1.

-pu (unités)

Facultatif. Unités de temps de l'intervalle d'interrogation du moniteur. Si vous spécifiez le paramètre `-pu`, vous devez également spécifier le paramètre `-pi`. La valeur par défaut de `-pu` est minutes. Indiquez l'une des options suivantes :

secondes

minutes

heures

jours

-tr

Facultatif. Indique la condition de déclenchement qui doit être satisfaite pour que la tâche définie ait lieu. Si la condition n'est pas satisfaite, selon l'agent source, la tâche de surveillance (par exemple, le transfert de fichiers) n'est pas démarrée. Une condition de déclenchement se compose de deux parties facultatives, condition et modèle, séparées par une virgule. Indiquez l'un des formats suivants:

- `condition,pattern`

où `condition` est l'une des valeurs suivantes:

correspond à

Pour chaque déclencheur qui est satisfait, la tâche définie est exécutée. `match` est la valeur par défaut.

Par exemple, si la correspondance est `*.go` et que les fichiers `LONDON.go` et `MANCHESTER.go` sont présents, la tâche est effectuée pour `LONDON.go` et une autre tâche est effectuée pour `MANCHESTER.go`.

Si le même fichier de déclenchement est présent à partir d'une interrogation précédente (c'est-à-dire que le fichier n'a pas été modifié), ce fichier a une condition de déclenchement non satisfaite. En d'autres termes, le fichier de déclenchement de correspondance doit être nouveau et avoir été modifié depuis la dernière interrogation avant l'exécution de la tâche définie.

noMatch

Aucun fichier du répertoire contrôlé ne correspond au modèle. En d'autres termes, si *l'un* des fichiers du répertoire contrôlé n'existe pas, la condition est satisfaite. Si aucun fichier ne correspond à la condition de déclenchement lors de la création du moniteur, celui-ci démarre instantanément, mais ne redémarre pas tant qu'une correspondance de fichier n'est pas trouvée, puis supprimée.

noSizeChange =n

Un minimum de l'un des fichiers du répertoire correspond au modèle et a une taille de fichier qui ne change pas pour les intervalles d'interrogation *n*. La valeur de *n* est un entier positif.

fileSize> =taille

Un minimum de l'un des fichiers du répertoire correspond au modèle et a une taille de fichier minimale supérieure ou égale à *taille*. La valeur *size* est une combinaison d'un entier avec une unité de taille facultative de B, Ko, Mo ou Go. Par exemple, `fileSize">"=10KB`. Si vous n'indiquez pas d'unité de taille, la taille par défaut utilisée est le nombre d'octets. Sur tous les systèmes d'exploitation, vous devez placer le symbole "supérieur à" (>) entre guillemets lorsque vous spécifiez l'option `fileSize` sur la ligne de commande, comme illustré dans cet exemple.

Le modèle est une séquence de correspondance de modèle de fichier au format générique ou d'expression régulière Java. La valeur par défaut du modèle est `*`, ou correspond à n'importe quel fichier, et le format par défaut est le format générique. Utilisez **-pt** pour spécifier le format du modèle.

Par exemple, la condition de déclenchement suivante est satisfaite lorsqu'un fichier existe dans le répertoire contrôlé avec le suffixe `.go`.

```
-tr match,*.go
```

La condition de déclenchement suivante est satisfaite lorsqu'aucun fichier du répertoire contrôlé ne porte le suffixe `.stop`.

```
-tr noMatch,*.stop
```

Vous ne pouvez spécifier *condition, pattern* que si vous spécifiez également le paramètre **-md**.

```
condition
```

où *condition* est l'une des valeurs suivantes:

queueNotvide

La file d'attente surveillée n'est pas vide. En d'autres cas, si la file d'attente surveillée contient des messages IBM MQ, la condition est satisfaite. Une seule tâche est exécutée pour tous les messages de la file d'attente.

completeGroups

Il existe un groupe complet dans la file d'attente surveillée. C'est-à-dire que si *l'un* des groupes de messages IBM MQ de la file d'attente surveillée est terminé, la condition est remplie. Une tâche individuelle est exécutée pour chaque groupe complet de la file d'attente.

Si un message unique qui ne se trouve pas dans un groupe est placé dans la file d'attente, il est traité comme s'il s'agissait d'un groupe complet et une tâche est exécutée pour le message unique.

Vous ne pouvez spécifier *condition* que si vous spécifiez également le paramètre **-mq**.

Pour chaque moniteur que vous créez, vous ne pouvez spécifier le paramètre **-tr** qu'une seule fois.

-tc

Facultatif. Indique que le fichier déclenché contient un ou plusieurs chemins de fichier pour générer une demande de transfert. Le format par défaut du contenu du fichier de déclenchement est une entrée de fichier sur chaque ligne. Indiquez les chemins d'accès au fichier en tant que *chemin d'accès au fichier source* ou *chemin d'accès au fichier source, chemin d'accès au fichier cible*. Ce paramètre est disponible uniquement pour les déclencheurs de moniteur de répertoire `match` et `noSizeChange`.

-tcr (modèle)

Facultatif. Indique une expression régulière de remplacement pour l'analyse syntaxique des fichiers de déclenchement. Si vous spécifiez le paramètre **-tcr**, vous devez également spécifier le paramètre **-tc**.

Concevez le modèle pour analyser chaque entrée de ligne complètement avec un ou deux groupes de capture. Le groupe 1 définit le chemin d'accès au fichier source et le groupe facultatif 2 définit le chemin d'accès au fichier cible. Il s'agit du comportement par défaut, que vous pouvez modifier à l'aide du paramètre **-tcc**.

Pour plus d'informations et d'exemples, consultez [«Utilisation d'un fichier de déclenchement»](#), à la page 194.

-tcc

Facultatif. Définit l'ordre du groupe de capture de l'expression \n \ régulière.

srcDest

La valeur par défaut où le groupe 1 est le chemin d'accès au fichier source et le groupe 2 est le chemin d'accès au fichier cible.

destSrc

L'inverse de `srcDest`. Le groupe 1 est le chemin d'accès au fichier de destination et le groupe 2 est le chemin d'accès au fichier source. Assurez-vous que l'expression régulière de `destSrc` comporte deux groupes de capture.

Si vous spécifiez le paramètre **-tcc**, vous devez également spécifier le paramètre **-tcr**.

-x (modèle_exclusion)

Facultatif. Indique les fichiers qui sont exclus de la correspondance de modèle de déclencheur. Le modèle de déclencheur est spécifié par le paramètre **-tr**.

Le modèle est une séquence de correspondance de modèle de fichier au format générique ou d'expression régulière Java. Le format par défaut est le format générique. Utilisez le paramètre **-pt** pour spécifier le format du modèle.

-mmd (métadonnées du moniteur)

Facultatif. Indique les métadonnées définies par l'utilisateur qui sont transmises aux points d'exit du moniteur. Le paramètre peut prendre une ou plusieurs paires de noms séparées par des virgules. Chaque paire de noms se compose d'une valeur *name=*. Vous pouvez utiliser le paramètre **-mmd** plusieurs fois dans une commande.

-pt (type_canevas)

Facultatif. Type de canevas utilisé par les paramètres **-tr** et **-x**. Les valeurs admises sont :

génériques

Les modèles sont évalués en tant que modèles génériques. Un astérisque (*) correspond à zéro ou plusieurs caractères et un point d'interrogation (?) correspond exactement à un caractère. Il s'agit de l'option par défaut.

expression régulière

Les modèles sont évalués en tant qu'expressions régulières Java. Pour plus d'informations, voir la section [«Expressions régulières utilisées par MFT»](#), à la page 679.

-bs (matches_per_task)

Facultatif. Nombre maximal de correspondances de déclencheur à inclure dans une seule tâche. Par exemple, si la valeur 5 est spécifiée pour *matches_per_task* et que neuf correspondances de déclencheur se produisent dans un intervalle d'interrogation unique, deux tâches sont effectuées.

La première tâche correspond aux déclencheurs 1-5 inclus, et la deuxième tâche correspond aux déclencheurs 6-9. La valeur par défaut de `matches_per_task` est 1.

Le paramètre **-bs** est pris en charge uniquement lorsque le code XML de définition de tâche que vous fournissez au paramètre **-mt** est un `managedTransfer`. Un `managedCall` n'est pas pris en charge avec le paramètre **-bs**.

-mquserid (userID)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de files d'attente de commandes.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de commandes. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous êtes invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

-dv (variable_par_défaut)

Facultatif. Liste de variables par défaut, séparées par des virgules, pouvant être utilisées dans la substitution de variables lors de la surveillance d'une file d'attente. Les valeurs sont au format d'une paire clé-valeur. Exemple :

```
-dv size=medium,color=blue
```

Pour plus d'informations sur la substitution de variables, voir [«Personnalisation des tâches du moniteur de ressources MFT avec la substitution de variables»](#), à la page 185. Vous ne pouvez spécifier le paramètre **-dv** que si vous avez également spécifié le paramètre **-mq**.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

-p (options_configuration)

Facultatif. Ce paramètre détermine l'ensemble des options de configuration à utiliser pour annuler le transfert. Par convention, utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble de fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

Exemples

Dans cet exemple, un nouveau moniteur de ressources nommé MYMONITOR est créé à l'aide de l'agent de surveillance MYAGENT. A condition qu'un fichier de plus de 5 Mo soit présent dans le répertoire `C:\wmqfte\monitors`, le transfert de fichier défini dans le fichier `C:\templates\transfer_reports.xml` est démarré. MYAGENT est également l'agent source du transfert de fichiers défini dans `C:\templates\transfer_reports.xml`:

```
fteCreateMonitor -ma MYAGENT -md C:\wmqfte\monitors -mn MYMONITOR -mt C:\templates\transfer_reports.xml  
-tr fileSize">"=5MB,*go
```

Dans cet exemple, un moniteur de ressources appelé MONITOR1 à l'aide de l'agent AGENT1 est créé pour transférer des fichiers de plus de 5 Mo et est exporté vers le fichier XML `monitor.xml`.

```
fteCreateMonitor -ox monitor.xml -ma AGENT1 -mn MONITOR1 -mt task.xml -tr "fileSize>=5MB,*zip"
```

Le fichier XML est ensuite importé et modifié pour exclure les fichiers dont la taille est supérieure à 10MB.

```
fteCreateMonitor -ix monitor.xml -x "fileSize>=10MB,*zip" -f
```

Dans cet exemple, un nouveau moniteur de ressources est créé sous le nom MYMONITOR à l'aide de l'agent MYAGENT.

```
fteCreateMonitor -ma MYAGENT -md c:\wmqfte -mn MYMONITOR -mt c:\templates\transfer_reports.xml -tr "fileSize>=5MB,*go"
```

Toutefois, le déclencheur est initialement défini de manière incorrecte pour surveiller c : \wmqfte plutôt que c : \wmqfte\monitors. La demande **fteCreateMonitor** est immédiatement émise à nouveau avec le répertoire du moniteur corrigé et les paramètres **-f** (remplacement) et **-c** (effacement de l'historique) utilisés pour mettre à jour le moniteur.

```
fteCreateMonitor -ma MYAGENT -md c:\wmqfte\monitors -mn MYMONITOR -mt c:\templates\transfer_reports.xml -tr "fileSize>=5MB,*go" -f -c
```

Codes retour

Code retour	Description
0	La commande a abouti.
1	La commande a échoué.

fteCreateTemplate: créer un nouveau modèle de transfert de fichiers

La commande **fteCreateTemplate** crée un modèle de transfert de fichier que vous pouvez conserver pour une utilisation ultérieure. Le seul paramètre obligatoire est le paramètre **-tn** (*template_name*). Tous les autres paramètres sont facultatifs, bien que si vous spécifiez une spécification de fichier source, vous devez également fournir un fichier de destination. De même, si vous spécifiez un fichier de destination, vous devez également spécifier une spécification de fichier source.

Objet

Utilisez la commande **fteCreateTemplate** pour créer un modèle de transfert de fichiers qui stocke vos détails de transfert jusqu'à ce que vous souhaitiez les utiliser ultérieurement. Utilisez des modèles de transfert pour stocker les paramètres de transfert de fichiers communs pour les transferts répétés ou complexes. Une fois que vous avez créé un modèle de transfert, soumettez le modèle à l'aide de IBM MQ Explorer. Vous ne pouvez pas soumettre de modèle de transfert à partir de la ligne de commande.

Le modèle de transfert que vous créez à l'aide de la commande **fteCreateTemplate** n'est pas identique au message XML que vous créez à l'aide du paramètre **-gt** de la commande **fteCreateTransfer**. Vous ne pouvez pas utiliser les deux types de modèle différents de manière interchangeable.

Vous pouvez exécuter la commande **fteCreateTemplate** à partir de n'importe quel système pouvant se connecter au réseau IBM MQ, puis l'acheminer vers le gestionnaire de files d'attente de coordination. Spécifiquement pour que la commande s'exécute, vous devez avoir installé Managed File Transfer sur ce système et avoir configuré le composant Managed File Transfer sur ce système pour communiquer avec le réseau IBM MQ.

Cette commande utilise le fichier `command.properties` pour se connecter au gestionnaire de files d'attente de commandes pour la topologie Managed File Transfer. Si le fichier `command.properties` contient la propriété **connectionQMgrHost**, la commande se connecte au gestionnaire de files d'attente de commandes à l'aide du transport CLIENT. Sinon, la commande se connecte au gestionnaire de files d'attente de commandes à l'aide du transport BINDINGS. Si le fichier `command.properties` n'existe pas, la commande échoue et génère l'erreur suivante:

```
BFGCL0491E: Missing or corrupt command.properties file. Use the fteSetupCommands command to correct this condition. Additional information might be contained in this exception BFGUB0009E: The following required property file is missing: "MQ_DATA_PATH\mqft\coordination\coordination_qmgr_name\command.properties"
```

Pour plus d'informations, voir [Le fichier command.properties](#) ..

Vous pouvez spécifier plusieurs fichiers source pour un transfert de fichier mais un seul agent de destination ; le transfert d'un fichier vers plusieurs agents de destination n'est pas pris en charge.

Toutefois, vous pouvez transférer plusieurs fichiers source vers plusieurs fichiers cible sur un même agent cible.

Pour savoir comment transférer des fichiers, voir [«Instructions pour le transfert de fichiers»](#), à la page 648.

Caractères spéciaux

Soyez prudent lorsque vous utilisez des paramètres qui contiennent des caractères spéciaux afin d'éviter que le shell de commandes n'interprète les caractères d'une manière que vous n'attendez pas.

z/OS Par exemple, les noms de fichiers qualifiés complets qui contiennent des guillemets simples et les spécifications source qui contiennent des astérisques peuvent être interprétés par le shell de commandes plutôt que transmis dans la demande de transfert. Pour éviter que des caractères soient interprétés par le shell de commandes, placez le paramètre entier entre guillemets doubles, comme indiqué dans les deux derniers exemples [«Exemples»](#), à la page 446, ou mettez en échappement les caractères spéciaux à l'aide de la séquence d'échappement du shell de commandes.

chemins relatifs

La commande **fteCreateTemplate** prend en charge l'utilisation de chemins de fichier relatifs.

Sur les systèmes répartis **z/OS** et z/OS UNIX System Services par défaut, les chemins d'accès sont considérés comme relatifs au répertoire de base de l'utilisateur sous lequel l'agent s'exécute. Pour modifier le répertoire par rapport auquel les noms de chemin sont évalués, définissez la propriété `transferRoot` dans le fichier `agent.properties`. Ce fichier se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr/agents/agent_name`. Ajoutez la ligne suivante au fichier :

```
transferRoot=directory_name
```

Vous devez mettre en échappement les chemins Windows ou les écrire au format UNIX. Par exemple, spécifiez `C:\TransferRoot` sous la forme `C:\\TransferRoot` ou `C:/TransferRoot`.

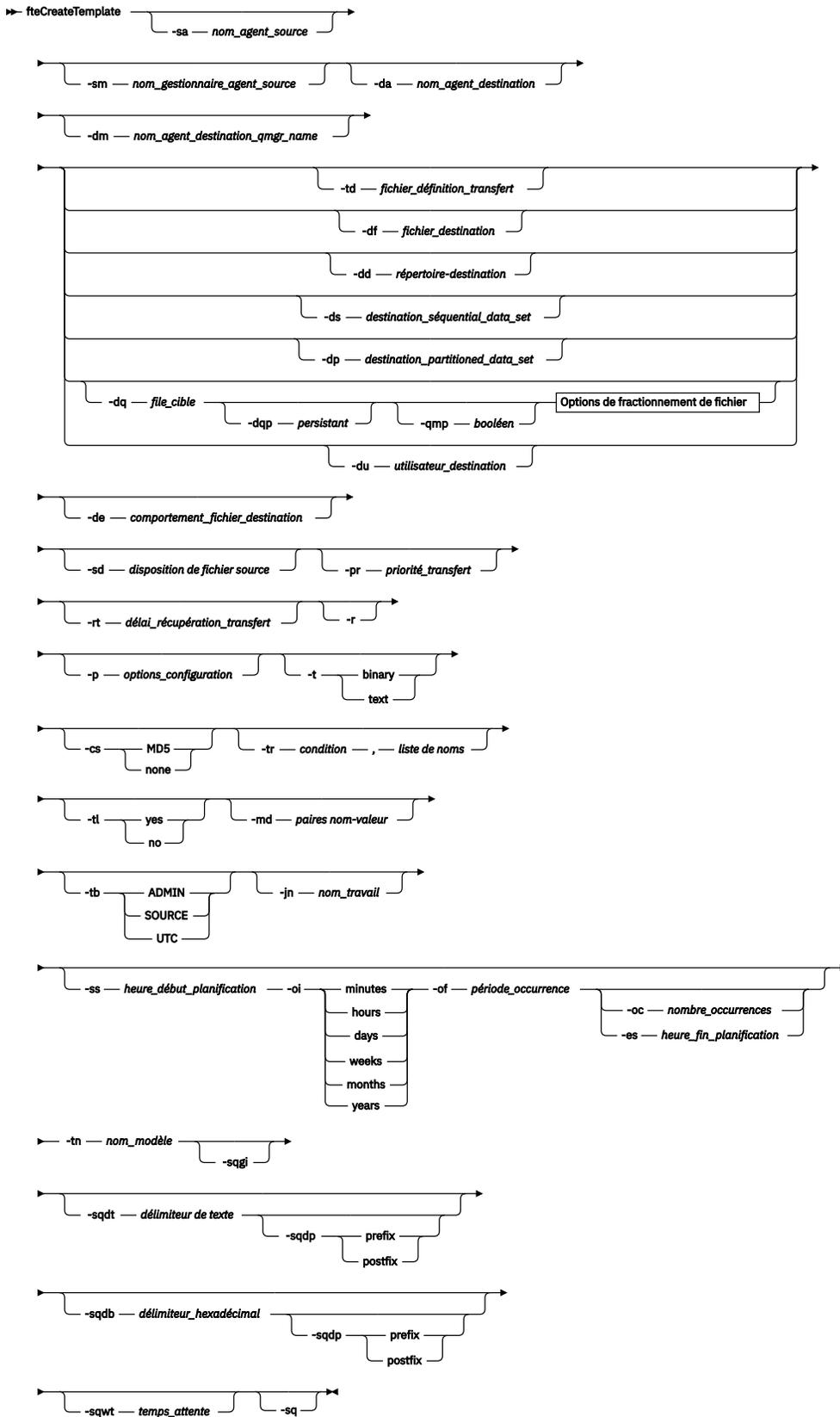
z/OS Sous z/OS, par défaut, le nom d'utilisateur sous lequel l'agent s'exécute est ajouté en tant que préfixe de qualificatif de haut niveau aux spécifications de fichier qui n'ont pas été entièrement qualifiées. Par exemple : `//ABC.DEF`. Pour modifier la valeur ajoutée en tant que préfixe au nom de fichier, définissez la propriété `HLQ transferRoot` dans le fichier `agent.properties`. Ce fichier se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr/agents/agent_name`. Ajoutez la ligne suivante au fichier :

```
transferRootHLQ=prepend_value
```

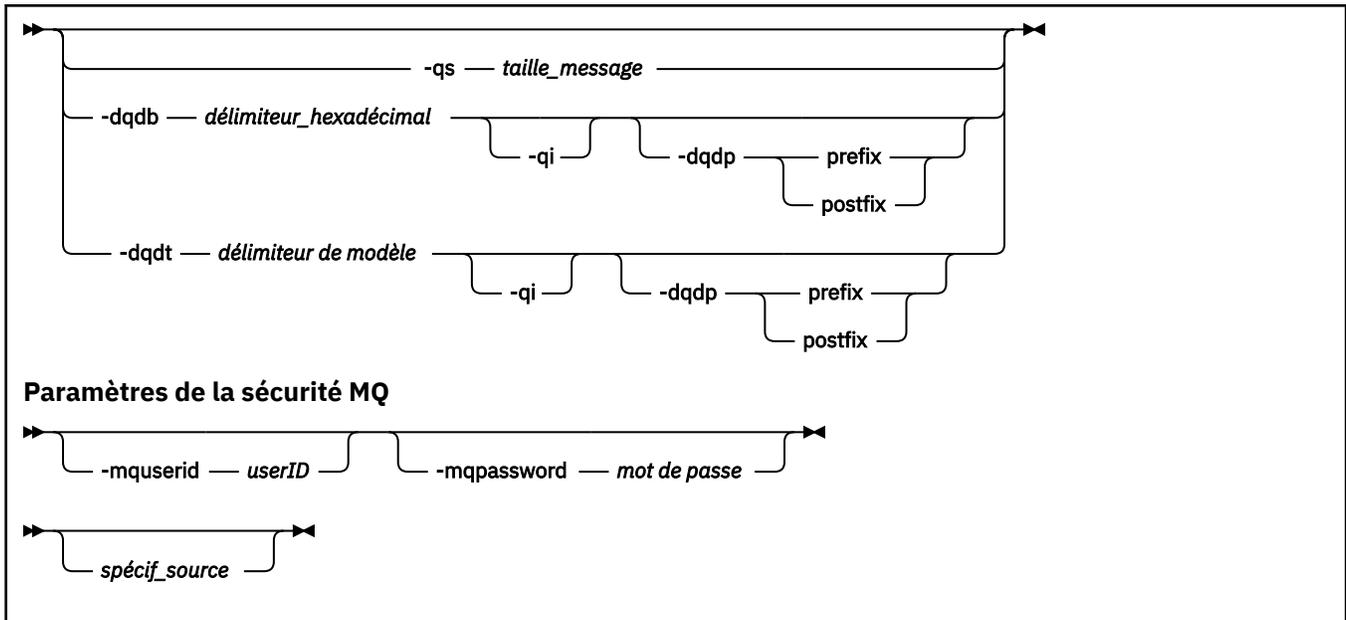
z/OS Toutefois, pour les transferts impliquant un noeud Connect:Direct sur un système z/OS, la spécification de l'ensemble de données est interprétée comme un nom qualifié complet. Aucun qualificatif de haut niveau n'est ajouté au nom de fichier.

Syntaxe

fteCreateTemplate



Options de fractionnement de fichier



Paramètres

-sa nom_agent_source

Facultatif. Nom de l'agent à partir duquel le fichier source est transféré. Si vous ne spécifiez pas ce nom d'agent lorsque vous créez le modèle, vous devez spécifier le nom de l'agent source lorsque vous utilisez le modèle.

-sm nom_gestionnaire_agent_source

Facultatif. Nom du gestionnaire de files d'attente auquel l'agent source est connecté.

Si vous ne spécifiez pas le paramètre **-sm**, le gestionnaire de files d'attente utilisé est déterminé par l'ensemble des options de configuration utilisées, en fonction du nom de l'agent source. Si le nom du gestionnaire de files d'attente ne peut pas être déterminé à l'aide de ces options, la création du modèle de transfert échoue. Par exemple, la création du modèle échoue si le fichier `agent.properties` de l'agent source est introuvable.

-da nom_agent_destination

Facultatif. Nom de l'agent vers lequel le fichier est transféré. Si vous ne spécifiez pas le nom de l'agent de destination lorsque vous créez le modèle, vous devez spécifier le nom de l'agent de destination lorsque vous utilisez le modèle.

-dm nom_gestionnaire_agent_destination

Facultatif. Nom du gestionnaire de files d'attente auquel l'agent de destination est connecté.

Si vous ne spécifiez pas le paramètre **-dm**, le gestionnaire de files d'attente utilisé est déterminé par l'ensemble des options de configuration utilisées, en fonction du nom de l'agent de destination. Si le nom du gestionnaire de files d'attente ne peut pas être déterminé à l'aide de ces options, la création du modèle de transfert échoue. Par exemple, la création du modèle échoue si le fichier `agent.properties` de l'agent de destination est introuvable.

-td fichier_définition_transfert

Facultatif. Nom du document XML qui définit une ou plusieurs spécifications de fichier source et de destination pour le transfert.

L'un des paramètres **-td**, **-df**, **-dd**, **-ds**, **-dq**, **-duet** ou **-dp** est requis. Si vous spécifiez le paramètre **-td**, vous ne pouvez pas spécifier les fichiers source, ni spécifier les paramètres **-df**, **-dd**, **-ds**, **-dp**, **-dq**, **-du**, **-sd**, **-r**, **-de**, **-tou** ou **-cs**.

La commande **fteCreateTemplate** localise le fichier de définition de transfert par rapport à votre répertoire en cours. Si vous ne pouvez pas utiliser la notation de chemin relatif pour spécifier l'emplacement du fichier de définition de transfert, utilisez le chemin d'accès complet et le nom de fichier du fichier de définition de transfert à la place.

z/OS Sous z/OS, vous devez stocker le fichier de définition de transfert dans un fichier UNIX sur z/OS UNIX System Services. Vous ne pouvez pas stocker les fichiers de définition de transfert dans des fichiers séquentiels z/OS ou des membres PDS.

IBM i Sous IBM i, vous devez stocker le fichier de définition de transfert dans le système de fichiers intégré.

Pour plus d'informations, voir [Utilisation des fichiers de définition de transfert](#).

-df fichier_destination

Facultatif. Nom du fichier de destination. Indiquez un nom de fichier valide sur le système sur lequel l'agent cible est exécuté.

Si l'agent de destination est un agent de pont Connect:Direct, le fichier de destination est spécifié au format *connect_direct_node_name:file_path*. L'agent de pont Connect:Direct accepte uniquement les chemins de fichier spécifiés dans ce format. **z/OS** Si l'agent de destination est un agent de pont Connect:Direct et que la destination est un membre PDS, vous devez également spécifier le paramètre **-de** avec la valeur *overwrite*.

L'un des paramètres **-td**, **-df**, **-dd**, **-ds**, **-dq**, **-duet** **-dp** est requis. Si vous spécifiez le paramètre **-df**, vous ne pouvez pas spécifier les paramètres **-td**, **-dd**, **-dp**, **-dq**, **-duou** **-ds** car ces paramètres s'excluent mutuellement.

-dd répertoire_destination

Facultatif. Nom du répertoire dans lequel le fichier est transféré. Indiquez un nom de répertoire valide sur le système sur lequel l'agent de destination est en cours d'exécution.

Si l'agent de destination est un agent de pont Connect:Direct, le répertoire de destination est spécifié au format *connect_direct_node_name:directory_path*. Si l'agent de destination est un agent de pont Connect:Direct et que la destination est un PDS, vous devez également spécifier le paramètre **-de** avec la valeur *overwrite*.

L'un des paramètres **-td**, **-df**, **-dd**, **-ds**, **-dq**, **-duet** **-dp** est requis. Si vous spécifiez le paramètre **-dd**, vous ne pouvez pas spécifier les paramètres **-td**, **-df**, **-dp**, **-dq**, **-duou** **-ds** car ces paramètres s'excluent mutuellement.

z/OS -ds paramètre_données_séquentiel_destination

z/OS seulement. Facultatif. Nom du fichier séquentiel ou du membre PDS dans lequel les fichiers sont transférés. Indiquez un nom de fichier séquentiel ou un membre de fichier partitionné.

L'un des paramètres **-td**, **-df**, **-dd**, **-ds**, **-dq**, **-duet** **-dp** est requis. Si vous spécifiez le paramètre **-ds**, vous ne pouvez pas spécifier les paramètres **-td**, **-dd**, **-df**, **-dq**, **-duou** **-dp** car ces paramètres s'excluent mutuellement.

La syntaxe du nom de fichier est la suivante:

```
//data_set_name{;attribute;..;attribute}
```

ou

```
//pds_data_set_name(member_name){;attribute;..;attribute}
```

En d'autres termes, un spécificateur de nom de fichier préfixé avec // et éventuellement suivi d'un certain nombre d'attributs séparés par des points-virgules.

Si le jeu de données se trouve sur un noeud Connect:Direct , vous devez préfixer le nom du jeu de données avec le nom du noeud. Exemple :

```
CD_NODE1:/' OBJECT.LIB' ;RECFM(F,B);BLKSIZE(800);LRECL(80)
```

Si l'agent de destination est un agent de pont Connect:Direct et que la destination est un membre PDS, vous devez également spécifier le paramètre **-de** avec la valeur `overwrite`. Pour plus d'informations sur les transferts de fichiers vers ou depuis des noeuds Connect:Direct , voir «[Transfert de fichiers vers et depuis des noeuds Connect:Direct](#)», à la page 653.

Pour les transferts impliquant uniquement des agents Managed File Transfer , si la partie nom de fichier est placée entre apostrophes, elle indique un nom de fichier qualifié complet. Si le nom de fichier n'est pas encadré par des guillemets simples, le système ajoute le qualificatif de haut niveau par défaut pour l'agent de destination (soit la valeur de la propriété d'agent `HLQ transferRoot`, soit l'ID utilisateur sous lequel l'agent s'exécute, si vous n'avez pas défini le qualificatif de haut niveau `transferRoot`).

Remarque :  Toutefois, pour les transferts impliquant un noeud Connect:Direct sur un système z/OS , la spécification de l'ensemble de données est interprétée comme un nom qualifié complet. Aucun qualificatif de haut niveau n'est ajouté au nom de fichier. C'est le cas même si le nom de fichier est encadré par des guillemets simples.

Les attributs de fichier sont utilisés pour créer un fichier ou pour s'assurer qu'un fichier existant est compatible. La spécification des attributs de fichier est dans un format adapté à BPXWDYN (voir [Demande d'allocation dynamique](#) pour plus d'informations). Lorsque l'agent doit créer un fichier de destination, les attributs BPXWDYN suivants sont automatiquement spécifiés: DSN (*nom_fichier*) NEW CATALOG MSG (*numeric_file_descriptor*), où *numeric_file_descriptor* est un descripteur de fichier généré par Managed File Transfer. Pour un transfert de fichier à fichier, les attributs de RECFM, LRECL et BLKSIZE de la source sont sélectionnés pour un nouveau fichier de destination. Notez que le paramètre SPACE d'un nouveau fichier de destination n'est pas défini par Managed File Transfer et que les valeurs par défaut du système sont utilisées. Par conséquent, il est recommandé de spécifier l'attribut SPACE lorsqu'un nouvel ensemble de données doit être créé. Vous pouvez utiliser la propriété **bpxwdynAllocAdditionalProperties** dans le fichier `agent.properties` pour définir les options BPXWDYN qui s'appliquent à tous les transferts. Pour plus d'informations, voir «[Le fichier MFT agent.properties](#)», à la page 564.

Certaines options BPXWDYN ne doivent pas être spécifiées en cas d'utilisation de la commande **fteCreateTemplate**, de la commande **fteCreateTransfer** ou de la propriété **bpxwdynAllocAdditionalOptions** dans le fichier `agent.properties`. Pour obtenir la liste de ces propriétés, voir «[Propriétés BPXWDYN que vous ne devez pas utiliser avec MFT](#)», à la page 661.

Lorsque vous transférez un fichier ou un ensemble de données sur bande, tout ensemble de données existant déjà sur la bande est remplacé. Les attributs du nouvel ensemble de données sont définis à partir des attributs transmis dans la définition de transfert. Si aucun attribut n'est spécifié, les attributs sont définis sur la même valeur que l'ensemble de données source ou sur les valeurs par défaut lorsque la source est un fichier. Les attributs d'un fichier bande existant sont ignorés.

Le paramètre **-ds** n'est pas pris en charge lorsque l'agent de destination est un agent de pont de protocole.

-dp paramètre_données_partition_destination

z/OS seulement. Facultatif. Nom de l'ensemble de données partitionnées de destination dans lequel les fichiers sont transférés. Indiquez un nom de fichier partitionné. Si un PDS est créé à la suite du transfert, ce PDS est créé en tant que PDSE par défaut. Vous pouvez remplacer la valeur par défaut en spécifiant `DSNTYPE=PDS`.

L'un des paramètres **-td**, **-df**, **-dd**, **-ds**, **-dq**, **-duet** **-dp** est requis. Si vous spécifiez le paramètre **-dp** , vous ne pouvez pas spécifier les paramètres **-td**, **-dd**, **-df**, **-dq**, **-duou** **-ds** car ces paramètres s'excluent mutuellement.

La syntaxe du nom de fichier PDS est la suivante:

```
//pds_data_set_name{;attribute;..;attribute}
```

La syntaxe du nom de fichier est la même que celle décrite pour le paramètre **-ds** (*destination_sequential_data_set*). Tous les détails de syntaxe pour la spécification des ensembles de données qui se trouvent sur les noeuds Connect:Direct s'appliquent également au paramètre **-dp**. Si l'agent de destination est un agent de pont Connect:Direct, vous devez également spécifier le paramètre **-de** avec la valeur *overwrite*.

Le paramètre **-dp** n'est pas pris en charge lorsque l'agent de destination est un agent de pont de protocole.

-du utilisateur_destination

Facultatif. Nom de l'utilisateur dont l'espace fichier cible dans lequel les fichiers sont transférés. :

L'un des paramètres **-td**, **-df**, **-dd**, **-ds**, **-dp**, **-duet** **-dq** est requis. Si vous spécifiez le paramètre **-du**, vous ne pouvez pas spécifier les paramètres **-td**, **-dd**, **-df**, **-dp**, **-dq** ou **-ds** car ces paramètres s'excluent mutuellement.

Le paramètre **-du** n'est pas pris en charge lorsque l'agent de destination est un agent de pont de protocole ou un agent de pont Connect:Direct.

-dq file_destination

Facultatif. Nom d'une file d'attente de destination vers laquelle les fichiers sont transférés. Vous pouvez éventuellement inclure un nom de gestionnaire de files d'attente dans cette spécification, en utilisant le format *QUEUE@QUEUEMANAGER*. Si vous ne spécifiez pas de nom de gestionnaire de files d'attente, le nom du gestionnaire de files d'attente de l'agent de destination est utilisé si vous n'avez pas défini la propriété d'agent de sortie *enableClusterQueueInputs* sur *true*. Si vous avez défini la propriété d'agent de sortie *enableClusterQueueInputs* sur *true*, l'agent de destination utilise des procédures de résolution IBM MQ standard pour déterminer où se trouve la file d'attente. Vous devez indiquer un nom de file d'attente valide existant sur le gestionnaire de files d'attente.

L'un des paramètres **-td**, **-df**, **-dd**, **-ds**, **-dp**, **-duet** **-dq** est requis. Si vous spécifiez le paramètre **-dq**, vous ne pouvez pas spécifier les paramètres **-td**, **-dd**, **-df**, **-dp**, **-du** ou **-ds** car ces paramètres s'excluent mutuellement.

Le paramètre **-dq** n'est pas pris en charge lorsque l'agent de destination est un agent de pont de protocole ou un agent de pont Connect:Direct, ou lorsque la spécification source est une file d'attente.

-dqp persistant

Facultatif. Indique si les messages écrits dans la file d'attente de destination sont persistants. Les options valides sont les suivantes:

conforme

Écrit les messages persistants dans la file d'attente de destination. Il s'agit de la valeur par défaut.

false

Écrit les messages non persistants dans la file d'attente de destination.

Qdef

La valeur de persistance provient de l'attribut *DefPersistence* de la file d'attente de destination.

Vous ne pouvez spécifier le paramètre **-dqp** que si vous avez également spécifié le paramètre **-dq**.

-qmp booléen

Facultatif. Indique si les propriétés de message IBM MQ sont définies pour le premier message écrit dans la file d'attente de destination par le transfert. Les options valides sont les suivantes:

conforme

Définit les propriétés de message sur le premier message créé par le transfert.

false

Ne définit pas les propriétés de message sur le premier message créé par le transfert. Il s'agit de la valeur par défaut.

Vous ne pouvez spécifier le paramètre **-qmp** que si vous avez également spécifié le paramètre **-dq**. Pour plus d'informations, voir [«Propriétés de message MQ définies par MFT sur les messages écrits dans les files d'attente de destination»](#), à la page 698

-qs taille_message

Facultatif. Indique si le fichier doit être fractionné en plusieurs messages de longueur fixe. Tous les messages ont le même ID groupe IBM MQ ; le dernier message du groupe a l'indicateur IBM MQ LAST_MSG_IN_GROUP défini. La taille des messages est spécifiée par la valeur de *message_size*. Le format de *message_size* est *longueurunités*, où *longueur* est un entier positif et *unités* est l'une des valeurs suivantes:

B

octets. La valeur minimale autorisée est deux fois la valeur maximale en octets par caractère de la page de codes des messages de destination.

K

Cela équivaut à 1024 octets.

M

Cela équivaut à 1048576 octets.

Si vous spécifiez la valeur `text` pour le paramètre **-t** et que le fichier est un jeu de caractères codés sur deux octets ou un jeu de caractères codés sur plusieurs octets, il est scindé en messages sur la limite de caractères la plus proche de la taille de message spécifiée.

Vous ne pouvez spécifier le paramètre **-qs** que si vous avez également spécifié le paramètre **-dq**. Vous ne pouvez spécifier qu'un seul des paramètres **-qs**, **-dqdbet** **-dqdt**.

-dqdb délimiteur_hexadécimal

Facultatif. Indique le délimiteur hexadécimal à utiliser lors du fractionnement d'un fichier binaire en plusieurs messages. Tous les messages ont le même ID groupe IBM MQ ; le dernier message du groupe a l'indicateur IBM MQ LAST_MSG_IN_GROUP défini. Le format de spécification d'un octet hexadécimal comme délimiteur est `xNN`, où `N` est un caractère compris entre `0-9` et `a-f`. Vous pouvez spécifier une séquence d'octets hexadécimaux comme délimiteur en spécifiant une liste d'octets hexadécimaux séparés par des virgules, par exemple: `x3e , x20 , x20 , xbf`.

Vous ne pouvez spécifier le paramètre **-dqdb** que si vous avez également spécifié le paramètre **-dq** et que le transfert est en mode binaire. Vous ne pouvez spécifier qu'un seul des paramètres **-qs**, **-dqdbet** **-dqdt**.

-dqdt modèle

Facultatif. Indique l'expression régulière à utiliser lors du fractionnement d'un fichier texte en plusieurs messages. Tous les messages ont le même ID groupe IBM MQ ; le dernier message du groupe a l'indicateur IBM MQ LAST_MSG_IN_GROUP défini. Le format permettant de spécifier une expression régulière comme délimiteur est une expression régulière entre parenthèses, (*regular_expression*). La valeur de ce paramètre est évaluée en tant qu'expression régulière Java. Pour plus d'informations, voir la section [«Expressions régulières utilisées par MFT»](#), à la page 679.

Par défaut, la longueur de la chaîne à laquelle l'expression régulière peut correspondre est limitée par l'agent de destination à cinq caractères. Vous pouvez modifier ce comportement à l'aide de la propriété d'agent **maxDelimiterMatchLength**. Pour plus d'informations, voir [«Propriétés d'agent avancées»](#), à la page 565.

Vous ne pouvez spécifier le paramètre **-dqdt** que si vous avez également spécifié le paramètre **-dq** et la valeur `text` pour le paramètre **-t**. Vous ne pouvez spécifier qu'un seul des paramètres **-qs**, **-dqdbet** **-dqdt**.

-dqdp

Facultatif. Indique la position attendue du texte de destination et des délimiteurs binaires lors du fractionnement des fichiers. Vous ne pouvez spécifier le paramètre **-dqdp** que si vous avez également spécifié l'un des paramètres **-dqdt** et **-dqdb**.

Indiquez l'une des options suivantes :

préfixe

Les délimiteurs sont attendus au début de chaque ligne.

suffixe

Les délimiteurs sont attendus à la fin de chaque ligne. Il s'agit de l'option par défaut.

-qi

Facultatif. Indique si le délimiteur utilisé pour fractionner le fichier en plusieurs messages doit être inclus dans les messages. Si **-qi** est spécifié, le délimiteur est inclus à la fin du message qui contient les données de fichier précédant le délimiteur. Par défaut, le délimiteur n'est pas inclus dans les messages.

Vous ne pouvez spécifier le paramètre **-qi** que si vous avez également spécifié l'un des paramètres **-dqdt** et **-dqdb**.

-de comportement_fichier_destination

Facultatif. Indique l'action effectuée si un fichier de destination existe sur le système de destination. Les options valides sont les suivantes:

erreur

Signale une erreur et le fichier n'est pas transféré. Il s'agit de la valeur par défaut.

écrasement

Remplace le fichier cible existant.

Si vous spécifiez le paramètre **-de**, vous ne pouvez pas spécifier le paramètre **-td** car ces paramètres s'excluent mutuellement.

-sd disposition-fichier-source

Facultatif. Indique l'action effectuée sur un fichier source lorsque ce fichier source a été transféré vers sa destination. Les options valides sont les suivantes:

Quitter

Les fichiers source restent inchangés. Il s'agit de la valeur par défaut.

supprimer

Le fichier source est supprimé du système source une fois que le fichier source a été transféré.

 Sous z/OS, si la source est un fichier de bande et que vous spécifiez l'option `delete`, la bande est remontée pour supprimer le fichier. Ce comportement est dû au comportement de l'environnement système.

Si la source est une file d'attente et que vous spécifiez l'option `leave`, la commande renvoie une erreur et un transfert n'est pas demandé.

Si l'agent source est un agent de pont `Connect:Direct` et que vous spécifiez l'option `delete`, le comportement est différent du comportement d'élimination de la source habituel. L'un des cas suivants se produit:

- Si `Connect:Direct` utilise un processus généré par `Managed File Transfer` pour déplacer le fichier ou l'ensemble de données depuis la source, l'indication de l'option `delete` entraîne l'échec du transfert. Pour spécifier que le fichier source doit être supprimé, vous devez soumettre un processus `Connect:Direct` défini par l'utilisateur. Pour plus d'informations, voir la section «[Soumission d'un processus Connect:Direct défini par l'utilisateur à partir d'une demande de transfert de fichier](#)», à la page 245.
- Si `Connect:Direct` fait appel à un processus défini par l'utilisateur pour déplacer le fichier ou l'ensemble de données à partir de la source, ce paramètre est transmis au processus via la variable

symbolique intrinsèque **%FTEFDISP**. Le processus défini par l'utilisateur détermine si la source est supprimé. Le résultat du transfert dépend du résultat renvoyé par le processus défini par l'utilisateur.

Si vous spécifiez le paramètre **-sd**, vous ne pouvez pas spécifier le paramètre **-td** car ces paramètres s'excluent mutuellement. Toutefois, vous pouvez spécifier le comportement de disposition de la source dans le fichier de définition de transfert.

-pr priorité_transfert

Facultatif. Indique le niveau de priorité du transfert. La priorité est une valeur comprise entre 0 et 9, où 0 est la priorité la plus basse. Le niveau de priorité par défaut est 0 et, par défaut, le transfert utilise le niveau de priorité de l'agent source.

Cette valeur correspond à la valeur de priorité de message utilisée par IBM MQ. Pour plus d'informations, voir [Obtention de messages à partir d'une file d'attente: priorité](#). Le trafic des messages pour les données de transfert de fichiers est défini par défaut sur un niveau de priorité de 0, ce qui permet à votre trafic de messages IBM MQ d'être prioritaire.

V 9.0.1 -rt délai_récupération_transfert

Facultatif. Définit la durée, en secondes, pendant laquelle un agent source continue de tenter de récupérer un transfert de fichiers bloqué. Indiquez l'une des options suivantes :

-1

L'agent continue de tenter de reprendre le transfert bloqué jusqu'à ce que ce dernier soit terminé. Cette option est équivalente au comportement par défaut de l'agent si la propriété n'est pas définie.

0

L'agent arrête le transfert de fichiers dès qu'il entre dans la phase de reprise.

>0

L'agent continue de tenter de reprendre le transfert bloqué pendant le nombre de secondes défini par l'entier positif spécifié. Par exemple :

```
-rt 21600
```

indique que l'agent tente de récupérer le transfert pendant 6 heures à partir du moment où il entre en récupération. La valeur maximale de ce paramètre est 999999999.

La spécification de la valeur de délai d'attente de reprise de transfert de cette manière la définit par transfert. Pour définir une valeur globale pour tous les transferts dans un réseau Managed File Transfer, vous pouvez ajouter une propriété au [fichier agent.properties](#).

-p options_configuration

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour créer le modèle de transfert. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

-r

Facultatif. Transfert récursif des fichiers dans les sous-répertoires lorsque *source_specification* contient des caractères génériques. Lorsque Managed File Transfer est présenté avec un caractère générique comme *source_specification*, tous les répertoires qui correspondent au caractère générique sont transférés uniquement si vous avez spécifié le paramètre **-r**. Lorsque *source_specification* correspond à un sous-répertoire, tous les fichiers de ce répertoire et de ses sous-répertoires (y compris les fichiers masqués) sont toujours transférés.

Pour plus d'informations sur la façon dont Managed File Transfer gère les caractères génériques, voir [Utilisation de caractères génériques](#)

Si vous spécifiez le paramètre **-r**, vous ne pouvez pas spécifier le paramètre **-td** car ces paramètres s'excluent mutuellement. Toutefois, vous pouvez spécifier un comportement récursif dans le fichier de définition de transfert.

-t

Facultatif. Indique le type de transfert de fichier: mode binaire ou mode texte.

binaire

Les données du fichier sont transférées sans conversion. Il s'agit de la valeur par défaut.

texte

Les caractères de page de codes et de fin de ligne du fichier sont convertis. Les conversions exactes effectuées dépendent des systèmes d'exploitation de l'agent source et de l'agent cible.

z/OS Par exemple, la page de codes d'un fichier transféré de Windows à z/OS est convertie de ASCII en EBCDIC. Lorsqu'un fichier est converti d'ASCII en EBCDIC, les caractères de fin de ligne sont convertis des paires de caractères de retour chariot (CR) et de saut de ligne (LF) ASCII en caractères de nouvelle ligne (NL) EBCDIC.

z/OS Pour plus d'informations sur la façon dont les fichiers z/OS sont transférés, voir [Transfert de fichiers et de fichiers entre z/OS et des systèmes répartis](#) et [Transfert entre des fichiers](#).

Si vous spécifiez le paramètre **-t**, vous ne pouvez pas spécifier le paramètre **-td** car ces paramètres s'excluent mutuellement. Toutefois, vous pouvez spécifier le comportement du mode de transfert dans le fichier de définition de transfert.

-cs

Facultatif. Indique si un algorithme de total de contrôle est exécuté sur les données de transfert de fichier pour vérifier l'intégrité des fichiers transférés. Indiquez l'une des options suivantes :

MD5

Calcule un total de contrôle MD5 pour les données. Le total de contrôle résultant pour les fichiers source et cible est consigné dans le journal de transfert à des fins de validation. Par défaut, Managed File Transfer calcule des totaux de contrôle MD5 pour tous les transferts de fichiers.

aucun

Aucun total de contrôle MD5 n'est calculé pour les données de transfert de fichier. Les enregistrements du journal de transfert dont le total de contrôle a été défini sur aucun et la valeur du total de contrôle est vide. Exemple :

```
<checksum method="none"></checksum>
```

Si vous utilisez l'option `none`, vous pouvez améliorer les performances de transfert de fichier, en fonction de votre environnement. Toutefois, la sélection de cette option signifie qu'il n'y a pas de validation des fichiers source ou de destination.

Si vous spécifiez le paramètre **-cs**, vous ne pouvez pas spécifier le paramètre **-td** car ces paramètres s'excluent mutuellement. Toutefois, vous pouvez spécifier un comportement de total de contrôle dans le fichier de définition de transfert.

-tr

Facultatif. Indique une condition qui doit être vraie pour que ce transfert de fichier ait lieu. Si la condition n'est pas vérifiée, selon l'agent source, le transfert de fichiers est supprimé et aucun transfert n'a lieu. Indiquez le format suivant:

```
condition,namelist
```

où *condition* est l'une des valeurs suivantes:

file=existe

Il existe au moins un des fichiers de la liste de noms. En d'autres termes, si *l'un* des fichiers de la liste de noms existe, la condition est vraie.

fichier !=exist

Un minimum de l'un des fichiers de la liste de noms n'existe pas. En d'autres termes, si *l'un* des fichiers de la liste de noms n'existe pas, la condition est vraie.

filesize> =taille

Un minimum de l'un des fichiers de la liste de noms existe et sa taille minimale est spécifiée par *taille*. La valeur de *size* est un entier avec une unité de taille facultative de Ko, Mo ou Go. Par exemple, `filesize">"=10KB`. Si vous ne spécifiez pas d'unité de taille, la taille est supposée être en octets. Sur tous les systèmes d'exploitation, vous devez placer le symbole "supérieur à" (>) entre guillemets lorsque vous spécifiez l'option `filesize` sur la ligne de commande, comme illustré dans cet exemple.

Et où *namelist* est une liste séparée par des virgules de noms de fichiers situés sur le système source. Selon votre système d'exploitation, si vous souhaitez utiliser des noms de chemin ou des noms de fichier dans une liste de noms contenant des espaces, vous devrez peut-être les placer entre guillemets.

Vous pouvez spécifier plusieurs conditions de déclenchement en utilisant le paramètre **-tr** plusieurs fois. Toutefois, dans ce cas, chaque condition de déclenchement distincte doit être vraie pour que le transfert de fichiers ait lieu.

Remarque : Pour surveiller en permanence une ressource pour qu'une condition de déclenchement soit vraie, il est recommandé d'utiliser la surveillance des ressources. Vous pouvez créer un moniteur de ressources à l'aide de la commande `fteCreateMonitor` .

Dans l'exemple suivant, le fichier `file1.doc` est transféré de AGENT1 à AGENT2, à condition que le fichier `A.txt`, ou le fichier `B.txt`, ou les deux fichiers existent sur AGENT1 et que le fichier `A.txt`, ou le fichier `B.txt`, ou les deux fichiers soient égaux ou supérieurs à 1 Go:

```
fteCreateTemplate -tn JUPITER_AGENT_TRIGGER_TEST_TEMPLATE -sa AGENT1 -sm QM_JUPITER -da AGENT2 -dm
QM_NEPTUNE
-tr file=exist,C:\export\A.txt,C:\export\B.txt
-tr filesize">"=1GB,C:\export\A.txt,C:\export\B.txt
-df C:\import\file1.doc C:\export\file1.doc
```

Vous pouvez combiner des paramètres de déclenchement avec des paramètres de planification. Si vous spécifiez les deux types de paramètres, les conditions de déclenchement sont appliquées au transfert de fichiers créé par les paramètres de planification.

-tl

Facultatif. Indique si les échecs de déclenchement sont consignés. Indiquez l'une des options suivantes :

oui

Des entrées de journal sont créées pour les transferts déclenchés ayant échoué. Il s'agit du comportement par défaut même si vous ne spécifiez pas le paramètre **-tl** .

non

Aucune entrée de journal n'est créée pour les transferts déclenchés ayant échoué.

-md

Facultatif. Indique les métadonnées définies par l'utilisateur qui sont transmises aux points d'exit de l'agent. Le paramètre **-md** peut prendre une ou plusieurs paires nom-valeur séparées par des virgules. Chaque paire de noms se compose de *name=value*. Vous pouvez utiliser le paramètre **-md** plusieurs fois dans une commande.

 Sous z/OS, les espaces représentent des délimiteurs et vous devez donc utiliser des traits de soulignement pour séparer les valeurs. Par exemple, utilisez `kw=text1_text2_text3` au lieu de `kw="text1 text2 text3"`

-tb

Facultatif. Indique la base de temps à utiliser pour le transfert de fichiers planifié. C'est-à-dire si vous souhaitez utiliser une heure système ou un temps universel coordonné (UTC). Vous devez utiliser ce paramètre uniquement avec le paramètre **-ss**. Indiquez l'une des options suivantes :

admin

Les heures de début et de fin utilisées pour le transfert planifié sont basées sur l'heure et la date du système utilisé par l'administrateur. Il s'agit de la valeur par défaut.

source

Les heures de début et de fin utilisées pour le transfert planifié sont basées sur l'heure et la date du système où se trouve l'agent source.

Temps Universel Coordonné

Les heures de début et de fin utilisées pour le transfert planifié sont basées sur le temps universel coordonné (UTC).

-jn *nom_travail*

Facultatif. Identificateur de nom de travail défini par l'utilisateur qui est ajouté au message du journal lorsque le transfert a commencé.

-ss *heure_début_planification*

Facultatif. Indique l'heure et la date auxquelles vous souhaitez que le transfert planifié ait lieu. Utilisez l'un des formats suivants pour spécifier l'heure et la date. Indiquez l'heure à l'aide de l'horloge au format 24 heures:

```
yyyy-MM-ddThh:mm
```

```
hh:mm
```

Les transferts de fichiers planifiés démarrent dans la minute suivant l'heure de début de la planification, s'il n'y a aucun problème susceptible d'affecter le transfert. Par exemple, il peut y avoir des problèmes avec votre réseau ou votre agent qui empêchent le démarrage du transfert planifié.

-oi

Facultatif. Indique l'intervalle auquel se produit le transfert planifié. Vous devez utiliser ce paramètre uniquement avec le paramètre **-ss**. Indiquez l'une des options suivantes :

minutes

heures

jours

semaines

mois

années

-of *période_occurrence*

Facultatif. Indique la fréquence à laquelle le transfert planifié a lieu. Par exemple, toutes les **5** semaines ou tous les **2** mois. Vous devez spécifier ce paramètre avec les paramètres **-oi** et **-ss** uniquement. Si vous ne spécifiez pas ce paramètre, la valeur par défaut 1 est utilisée.

-oc *nombre_occurrences*

Facultatif. Indique le nombre de fois que vous souhaitez que ce transfert planifié ait lieu. Une fois le nombre d'occurrences atteint, le transfert planifié est supprimé.

Spécifiez ce paramètre avec les paramètres **-oi** et **-ss** uniquement.

Si vous spécifiez le paramètre **-oc**, vous ne pouvez pas spécifier le paramètre **-es** car ces paramètres s'excluent mutuellement.

Vous pouvez omettre les paramètres **-oc** et **-es** pour créer un transfert qui se répète indéfiniment.

-es heure_fin_planification

Facultatif. Date et heure de fin d'un transfert planifié répétitif.

Vous devez spécifier ce paramètre avec les paramètres **-oi** et **-ss** uniquement.

Si vous spécifiez le paramètre **-es**, vous ne pouvez pas spécifier le paramètre **-oc** car ces paramètres s'excluent mutuellement.

Vous pouvez omettre les paramètres **-es** et **-oc** pour créer un transfert qui se répète indéfiniment.

Utilisez l'un des formats suivants pour spécifier l'heure et la date de fin. Indiquez l'heure à l'aide de l'horloge au format 24 heures:

```
yyyy-MM-ddThh:mm  
hh:mm
```

-tn nom_modèle

Obligatoire. Nom du modèle que vous souhaitez créer. Utilisez une chaîne descriptive qui vous permet de sélectionner le modèle approprié pour les transferts à une date ultérieure. Il n'y a pas de limite spécifique à la longueur de cette chaîne, mais sachez que des noms trop longs peuvent ne pas s'afficher correctement dans certaines interfaces utilisateur.

Ne créez pas plusieurs modèles portant le même nom.

-sqgi

Facultatif. Indique que les messages sont regroupés par ID groupe IBM MQ. Le premier groupe complet est écrit dans le fichier de destination. Si ce paramètre n'est pas spécifié, tous les messages de la file d'attente source sont écrits dans le fichier de destination.

Vous ne pouvez spécifier le paramètre **-sqgi** que si vous avez également spécifié le paramètre **-sq**.

-sqdt délimiteur de texte

Facultatif. Indique une séquence de texte à insérer comme délimiteur lors de l'ajout de plusieurs messages à un fichier texte. Vous pouvez inclure des séquences d'échappement Java pour les littéraux chaîne dans le délimiteur. Par exemple, `-sqdt \u007d\n`.

Vous ne pouvez spécifier le paramètre **-sqdt** que si vous avez également spécifié le paramètre **-sq** et la valeur `text` pour le paramètre **-t**.

-sqdb délimiteur hexadécimal

Facultatif. Indique une ou plusieurs valeurs d'octet à insérer comme délimiteur lors de l'ajout de plusieurs messages à un fichier binaire. Chaque valeur doit être spécifiée sous la forme de deux chiffres hexadécimaux dans la plage 00-FF, précédés de `x`. Plusieurs octets doivent être séparés par des virgules. Par exemple, `-sqdb x08,xA4`.

Vous ne pouvez spécifier le paramètre **-sqdb** que si vous avez également spécifié le paramètre **-sq**. Vous ne pouvez pas spécifier le paramètre **-sqdb** si vous avez également spécifié la valeur `text` pour le paramètre **-t**.

-ppp

Facultatif. Indique la position d'insertion du texte source et des délimiteurs binaires. Vous ne pouvez spécifier le paramètre **-sqdp** que si vous avez également spécifié l'un des paramètres **-sqdt** et **-sqdb**.

Indiquez l'une des options suivantes :

préfixe

Les délimiteurs sont insérés au début de chaque message

suffixe

Les délimiteurs sont insérés à la fin de chaque message. Il s'agit de l'option par défaut.

-sqwt temps_attente

Facultatif. Indique le délai d'attente, en secondes, pour que l'une des conditions suivantes soit remplie:

- Pour qu'un nouveau message soit inséré dans la file d'attente
- Si le paramètre **-sqgi** a été spécifié, pour qu'un groupe complet soit placé dans la file d'attente

Si aucune de ces conditions n'est remplie dans le délai spécifié par *wait_time*, l'agent source arrête de lire dans la file d'attente et termine le transfert. Si le paramètre **-sqwt** n'est pas spécifié, l'agent source arrête immédiatement la lecture à partir de la file d'attente source si la file d'attente source est vide ou, dans le cas où le paramètre **-sqgi** est spécifié, s'il n'y a pas de groupe complet dans la file d'attente.

Vous ne pouvez spécifier le paramètre **-sqwt** que si vous avez également spécifié le paramètre **-sq**.

-sq

Facultatif. Indique que la source d'un transfert est une file d'attente.

-mquserid (userID)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de file d'attente de coordination.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de coordination. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

spécif_source

Obligatoire si vous avez spécifié l'un des paramètres **-df**, **-dd**, **-dp**, **-dpou** **-ds**. Si vous spécifiez le paramètre **-td**, ne spécifiez pas *source_specification*.

- Si vous n'avez pas spécifié le paramètre **-sq**, *source_specification* est une ou plusieurs spécifications de fichier qui déterminent la ou les sources du transfert de fichier. Les spécifications de fichier sont délimitées par des espaces. Les spécifications de fichier peuvent prendre l'une des cinq formes et inclure des caractères génériques. Pour plus d'informations sur les caractères génériques dans WMQFTE, voir «Utilisation de caractères génériques avec MFT», à la page 673. Vous pouvez mettre en échappement les astérisques qui font partie de la spécification de fichier en utilisant deux astérisques (**) dans la spécification de fichier.

Pour transférer des fichiers contenant des espaces dans leurs noms de fichier, placez les noms de fichier contenant des espaces entre guillemets. Par exemple, pour transférer le fichier a b.txt dans le fichier c d.txt, spécifiez le texte suivant dans le cadre de la commande

fteCreateTemplate :

```
-df "c d.txt" "a b.txt"
```

Chaque spécification de fichier doit être dans l'un des formats suivants:

Noms de fichiers

Nom d'un fichier, exprimé à l'aide de la notation appropriée pour le système sur lequel l'agent source est en cours d'exécution. Lorsqu'un nom de fichier est spécifié en tant que spécification de fichier source, le contenu du fichier est copié.

Répertoires

Nom d'un répertoire, exprimé à l'aide de la notation appropriée pour le système sur lequel l'agent source est en cours d'exécution. Lorsqu'un répertoire est spécifié en tant que spécification de fichier source, le contenu du répertoire est copié. Plus précisément, tous les fichiers du répertoire et de tous ses sous-répertoires, y compris les fichiers masqués, sont copiés.

Par exemple, pour copier le contenu de DIR1 dans DIR2 uniquement, indiquez DIR1/* DIR2

z/OS Fichier séquentiel

(z/OS uniquement). Nom d'un fichier séquentiel ou d'un membre de fichier partitionné. Dénoter les ensembles de données en faisant précéder le nom de l'ensemble de données de deux barres obliques (/).

z/OS Fichier partitionné

(z/OS uniquement). Nom d'un fichier partitionné. Dénoter les noms de fichier en les faisant précéder de deux barres obliques (/).

Nom de fichier ou répertoire sur un noeud Connect:Direct

(agent de pont Connect:Direct uniquement). Nom d'un noeud Connect:Direct, un signe deux-points (:) et un chemin de fichier ou de répertoire sur le système qui héberge le noeud Connect:Direct. Par exemple, *connect_direct_node_name:file_path*.

Si l'agent source est un agent de pont Connect:Direct, il n'accepte que les spécifications de source sous cette forme.

Remarque : Les caractères génériques ne sont pas pris en charge dans les chemins de fichier lorsque l'agent source est un agent de pont Connect:Direct.

- Si vous avez spécifié le paramètre **-sq**, *source_specification* est le nom d'une file d'attente locale sur le gestionnaire de files d'attente de l'agent source. Vous ne pouvez spécifier qu'une seule file d'attente source. La file d'attente source est spécifiée au format suivant:

```
QUEUE_NAME
```

Le nom du gestionnaire de files d'attente n'est pas inclus dans la spécification de file d'attente source car le gestionnaire de files d'attente doit être identique au gestionnaire de files d'attente de l'agent source.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemples

Dans cet exemple, un modèle de transfert appelé `payroll accounts monthly report template` est créé. Une fois soumis, ce modèle transfère tout fichier portant l'extension `.xls` de l'agent `PAYROLL1` vers l'agent `ACCOUNTS` dans les répertoires spécifiés:

```
fteCreateTemplate -tn "payroll accounts monthly report template" -sa PAYROLL -sm QM_PAYROLL1 -da ACCOUNTS -dm QM_ACCOUNTS -df C:\payroll_reports\*.xls C:\out\*.xls
```

Dans cet exemple, un modèle de transfert appelé `jupiter_neptune_sched_template` est créé. Une fois soumis, le modèle transfère le fichier `originalfile.txt` du système où se trouve `QM_JUPITER` vers le système où se trouve `QM_NEPTUNE`. Le transfert de fichiers est planifié pour avoir lieu à 09:00 en fonction de l'heure système du système où se trouve l'agent source et se produit toutes les deux heures quatre fois:

```
fteCreateTemplate -tn jupiter_neptune_sched_template -sa AGENT1 -sm QM_JUPITER -da AGENT2 -dm QM_NEPTUNE -tb source -ss 09:00 -oi hours -of 2 -oc 4 -df C:\import\transferredfile.txt C:\export\originalfile.txt
```

Dans cet exemple, un modèle de transfert appelé `jupiter neptune trigger template` est créé. Lorsque le modèle est soumis, le fichier `originalfile.txt` est transféré de `AGENT1` à `AGENT2`, à condition que le fichier `A.txt` existe sur `AGENT1`:

```
fteCreateTemplate -tn "jupiter neptune trigger template" -sa AGENT1 -sm QM_JUPITER -da AGENT2 -dm QM_NEPTUNE -tr file=exist,C:\export\A.txt -df C:\import\transferredfile.txt C:\export\originalfile.txt
```

z/OS Dans cet exemple, un modèle appelé `ascii_ebcdic_template` est créé. Lorsque le modèle est soumis, le fichier `originalfile.txt` est transféré du système où AGENT1 se trouve vers un ensemble de données `//USERID.TRANS.FILE.TXT` sur le système où se trouve AGENT2. Le mode texte a été sélectionné pour la conversion de données ASCII en EBCDIC.

```
fteCreateTemplate -tn ascii_ebcdic_template -t text -sa AGENT1 -da AGENT2
-ds "//TRANS.FILE.TXT;RECFM(V,B);BLKSIZE(6144);LRECL(1028);
SPACE(5,1)" C:\export\originalfile.txt
```

z/OS Dans cet exemple, un modèle appelé `ebcdic_ascii_template` est créé. Lorsque le modèle est soumis, un membre d'un fichier qualifié complet sur le système où se trouve AGENT1 est transféré vers un fichier sur le système où se trouve AGENT2. Le mode texte a été sélectionné pour convertir le fichier de EBCDIC en ASCII.

```
fteCreateTemplate -tn ebcdic_ascii_template -t text -sa AGENT1 -da AGENT2 -df /tmp/IEEUJV.txt
"//SYS1.SAMPLIB(IEEUJV)'"
```

Codes retour

Code retour	Description
0	La commande a abouti.
1	La commande a échoué.

fteCreateTransfer: démarrer un nouveau transfert de fichier

La commande **fteCreateTransfer** crée et démarre un nouveau transfert de fichier à partir de la ligne de commande. Cette commande peut démarrer un transfert de fichier immédiatement, planifier un transfert de fichier à une heure et une date ultérieures, répéter un transfert planifié une ou plusieurs fois et déclencher un transfert de fichiers sous certaines conditions.

Objet

Utilisez la commande **fteCreateTransfer** pour créer, puis démarrer un nouveau transfert de fichier à partir d'un agent Managed File Transfer.

Remarque : Les transferts de fichiers ne peuvent avoir lieu qu'entre les agents de la même topologie Managed File Transfer.

Pour savoir comment transférer des fichiers, voir [«Instructions pour le transfert de fichiers»](#), à la page 648. Pour la plateforme z/OS, vous pouvez transférer des fichiers texte, des fichiers et des groupes de données de génération (GDG).

Vous pouvez exécuter la commande **fteCreateTransfer** à partir de n'importe quel système pouvant se connecter au réseau IBM MQ, puis effectuer le routage vers le gestionnaire de files d'attente de l'agent source. En particulier, pour que la commande s'exécute, vous devez installer un composant Managed File Transfer (Service ou Agent) sur ce système et configurer le composant Managed File Transfer sur ce système pour qu'il communique avec le réseau IBM MQ.

Cette commande utilise un fichier de propriétés appelé `command.properties` pour se connecter au réseau IBM MQ. Si le fichier `command.properties` ne contient pas d'informations de propriété, une connexion en mode liaison est établie avec le gestionnaire de files d'attente par défaut sur le système local. Si le fichier `command.properties` n'existe pas, une erreur est générée. Pour plus d'informations, voir la section [«Le fichier MFT `command.properties`»](#), à la page 559.

Vous pouvez spécifier plusieurs fichiers source pour un transfert de fichiers, mais ils doivent provenir d'un seul agent source et se terminer à un seul agent cible. Le transfert d'un seul fichier source vers plusieurs fichiers de destination sur le même agent ou plusieurs agents différents n'est pas pris en charge dans

un même transfert. Le scriptage Ant peut être utilisé pour envoyer le même fichier source à plusieurs destinations sur un ou plusieurs agents. Pour plus d'informations, voir la section «[Utilisation d' Apache Ant avec MFT](#)», à la page 258.

Caractères spéciaux

Soyez prudent lorsque vous utilisez des paramètres qui contiennent des caractères spéciaux afin d'éviter que le shell de commandes n'interprète les caractères d'une manière que vous n'attendez pas. Par exemple, les noms de fichiers qualifiés complets qui contiennent des guillemets simples et les spécifications source qui contiennent des astérisques peuvent être interprétés par le shell de commandes plutôt que transmis dans la demande de transfert. Pour éviter que des caractères soient interprétés par le shell de commandes, placez le paramètre entier entre guillemets ou mettez en échappement les caractères spéciaux à l'aide de la séquence d'échappement du shell de commandes.

chemins relatifs

La commande **fteCreateTransfer** prend en charge l'utilisation de chemins de fichier relatifs. Pour les plateformes suivantes, par défaut, les chemins sont considérés comme étant relatifs au répertoire de base de l'utilisateur sous lequel l'agent s'exécute:

- **Multi** Multiplateformes
- **z/OS** UNIX System Services sous z/OS

Pour modifier le répertoire par rapport auquel les noms de chemin sont évalués, définissez la propriété `transferRoot` dans le fichier `agent.properties`. Ce fichier se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr/agents/agent_name`. Ajoutez la ligne suivante au fichier :

```
transferRoot=directory_name
```

Windows Par exemple, spécifiez `C:\TransferRoot` sous la forme `C:\\TransferRoot` ou `C:/TransferRoot`.

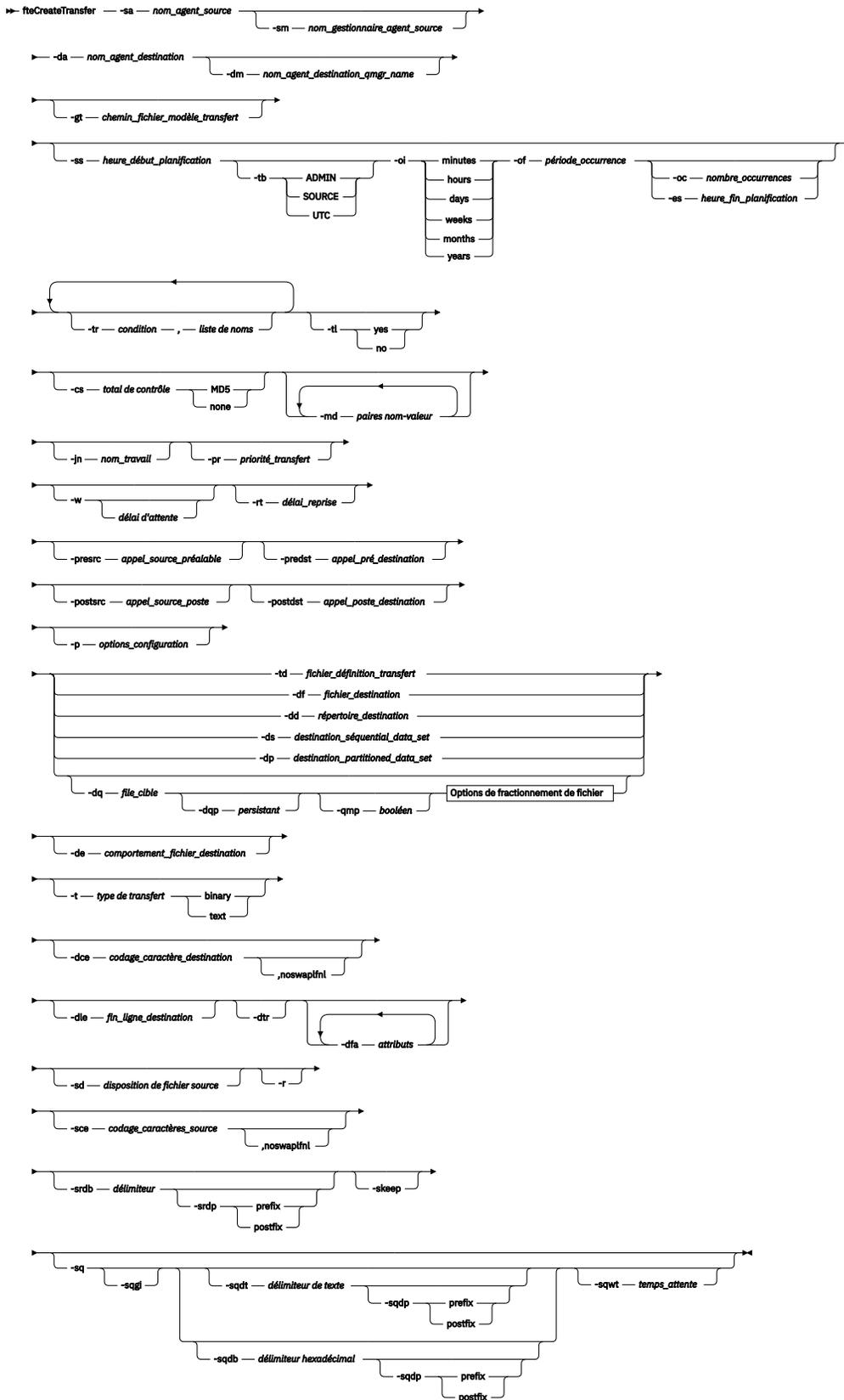
z/OS Sous z/OS, par défaut, le nom d'utilisateur sous lequel s'exécute l'agent est ajouté en tant que préfixe de qualificatif de haut niveau aux spécifications de fichier qui n'ont pas été entièrement qualifiées. Par exemple : `//ABC.DEF`. Pour modifier la valeur ajoutée en tant que préfixe au nom de fichier, définissez la propriété `HLQ transferRoot` dans le fichier `agent.properties`. Ce fichier se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr/agents/agent_name`. Ajoutez la ligne suivante au fichier :

```
transferRootHLQ=prepend_value
```

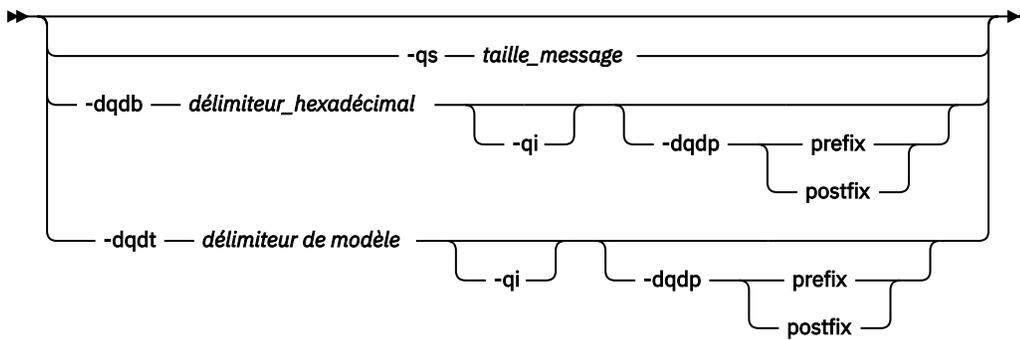
z/OS Toutefois, pour les transferts impliquant un noeud `Connect:Direct` sur un système z/OS, la spécification de l'ensemble de données est interprétée comme un nom qualifié complet. Aucun qualificatif de haut niveau n'est ajouté au nom de fichier.

Syntaxe

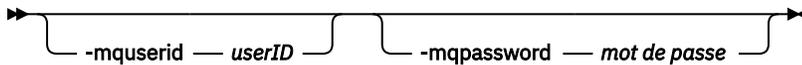
fteCreateTransfer



Options de fractionnement de fichier



Paramètres de la sécurité MQ



➤ *spécif_source* ➤

Paramètres de la spécification d'agent

-sa nom_agent_source

Obligatoire. Nom de l'agent à partir duquel les fichiers source sont transférés.

z/OS Si vous spécifiez un agent de pont de protocole comme agent source, vous ne pouvez pas spécifier un ensemble de données comme spécification de fichier source.

Si vous spécifiez le paramètre **-td** et que le fichier de définition de transfert contient l'agent source que vous souhaitez utiliser pour le transfert, n'indiquez pas le paramètre **-sa**.

-sm nom_gestionnaire_agent_source

Facultatif. Nom du gestionnaire de files d'attente auquel l'agent source est connecté.

Si vous ne spécifiez pas le paramètre **-sm**, le gestionnaire de files d'attente utilisé est déterminé par l'ensemble des options de configuration utilisées, qui est basé sur le nom de l'agent source. Si le fichier `agent.properties` de l'agent source est introuvable, le transfert de fichier échoue.

-da nom_agent_destination

Obligatoire. Nom de l'agent vers lequel les fichiers sont transférés.

Si vous spécifiez le paramètre **-td** et que le fichier de définition de transfert contient l'agent de destination que vous souhaitez utiliser pour le transfert, n'indiquez pas le paramètre **-da**.

-dm nom_gestionnaire_agent_destination

Facultatif. Nom du gestionnaire de files d'attente auquel l'agent de destination est connecté.

Si vous ne spécifiez pas le paramètre **-dm**, le gestionnaire de files d'attente utilisé est déterminé par l'ensemble des options de configuration en cours d'utilisation, qui est basé sur le nom de l'agent de destination. Si le fichier `agent.properties` de l'agent de destination est introuvable, le transfert de fichier échoue.

Paramètres de génération de modèles de transfert

-gt chemin_fichier_modèle_transfert

Facultatif. Génère un message XML de modèle de transfert et écrit ce message dans un fichier. Si vous spécifiez ce paramètre, aucune demande de transfert n'est envoyée à Managed File Transfer. A la place, le contenu du message de demande de transfert est écrit dans le document XML nommé. Vous pouvez ensuite utiliser ce document XML pour définir la tâche de surveillance des ressources. Pour plus d'informations sur la création d'un moniteur de ressources, voir [fteCreateMonitor command](#).

Si vous ne spécifiez pas ce paramètre, le comportement par défaut se produit et une demande de transfert réelle est effectuée.

Vous devez indiquer le chemin d'accès complet et le nom d'un fichier de sortie XML comme entrée pour ce paramètre, par exemple `C:\templates\transfer_reports.xml`

z/OS Sous z/OS, vous devez stocker le document de modèle de transfert dans un fichier UNIX sur z/OS UNIX System Services. Vous ne pouvez pas stocker des documents de modèle de transfert dans des fichiers séquentiels ou des membres PDS z/OS .

IBM i Sous IBM i, vous devez stocker le document de modèle de transfert dans le système de fichiers intégré.

Le message XML de modèle de transfert que vous créez à l'aide du paramètre **-gt** n'est pas le même que celui que vous créez à l'aide de la commande **fteCreateTemplate** , ce qui signifie que vous ne pouvez pas utiliser les deux types de modèle de manière interchangeable.

Remarque : Si vous souhaitez générer un document XML de modèle de transfert en exécutant la commande **fteCreateTransfer** avec le paramètre **-gt** , puis en fournissant ce document XML de modèle de transfert en tant qu'entrée à la commande **fteCreateTransfer** à l'aide du paramètre **-td** , vous devez vous assurer que le document XML de modèle de transfert a été généré en spécifiant les paramètres qui s'excluent mutuellement avec l'option **-td** .

Les paramètres mutuellement exclusifs de l'option **-td** sont les suivants:

- **-dd** *répertoire_destination*
- *Chemin source*
- **-df** *fichier_destination*
- **-cs** *total de contrôle*
- **-de** *comportement_fichier_destination*
- **-dq** *file_cible*
- **-t** *type de transfert*
- **-sd** *disposition de fichier source*

Par exemple, il n'est pas possible de spécifier à la fois les paramètres **-td** et **-t** (indiquant si le transfert est un transfert binaire ou texte) dans la commande **fteCreateTransfer** . Cela signifie que si vous souhaitez transmettre un document XML de modèle de transfert à la commande et spécifier que le transfert doit être un transfert de texte, vous devez créer le document XML en spécifiant les paramètres de texte **-gt** et **-t** .

Paramètres de planification des transferts

-ss *heure_début_planification*

Facultatif. Indique l'heure et la date auxquelles vous souhaitez que le transfert planifié ait lieu. Utilisez l'un des formats suivants pour spécifier l'heure et la date. Indiquez l'heure à l'aide de l'horloge au format 24 heures:

```
yyyy-MM-ddThh:mm
```

```
hh:mm
```

Les transferts de fichiers planifiés démarrent dans la minute suivant l'heure de début de la planification, s'il n'y a aucun problème susceptible d'affecter le transfert. Par exemple, il peut y avoir des problèmes avec votre réseau ou votre agent qui empêchent le démarrage du transfert planifié.

-tb

Facultatif. Indique la base de temps à utiliser pour le transfert de fichiers planifié. C'est-à-dire si vous souhaitez utiliser une heure système ou un temps universel coordonné (UTC). Vous devez utiliser ce paramètre uniquement avec le paramètre **-ss**. Indiquez l'une des options suivantes :

admin

Les heures de début et de fin utilisées pour le transfert planifié sont basées sur l'heure et la date du système utilisé par l'administrateur local. Il s'agit de la valeur par défaut.

source

Les heures de début et de fin utilisées pour le transfert planifié sont basées sur l'heure et la date du système où se trouve l'agent source.

Temps Universel Coordonné

Les heures de début et de fin utilisées pour le transfert planifié sont basées sur le temps universel coordonné (UTC).

-oi

Facultatif. Indique l'intervalle auquel se produit le transfert planifié. Vous devez utiliser ce paramètre uniquement avec le paramètre **-ss**. Indiquez l'une des options suivantes :

minutes

heures

jours

semaines

mois

années

-of période_occurrence

Facultatif. Indique la fréquence à laquelle le transfert planifié a lieu. Par exemple, toutes les **5** semaines ou tous les **2** mois. Vous devez spécifier ce paramètre avec les paramètres **-oi** et **-ss** uniquement. Si vous ne spécifiez pas ce paramètre, la valeur par défaut **1** est utilisée.

-oc nombre_occurrences

Facultatif. Indique le nombre de fois que vous souhaitez que ce transfert planifié ait lieu. Une fois le nombre d'occurrences atteint, le transfert planifié est supprimé.

Spécifiez ce paramètre avec les paramètres **-oi** et **-ss** uniquement.

Si vous spécifiez le paramètre **-oc**, vous ne pouvez pas spécifier le paramètre **-es** car ces paramètres s'excluent mutuellement.

Vous pouvez omettre les paramètres **-oc** et **-es** pour créer un transfert qui se répète indéfiniment.

-es heure_fin_planification

Facultatif. Date et heure de fin d'un transfert planifié répétitif.

Vous devez spécifier ce paramètre avec les paramètres **-oi** et **-ss** uniquement.

Si vous spécifiez le paramètre **-es**, vous ne pouvez pas spécifier le paramètre **-oc** car ces paramètres s'excluent mutuellement.

Vous pouvez omettre les paramètres **-es** et **-oc** pour créer un transfert qui se répète indéfiniment.

Utilisez l'un des formats suivants pour spécifier l'heure et la date de fin. Indiquez l'heure à l'aide de l'horloge au format 24 heures:

```
yyyy-MM-ddThh:mm
```

```
hh:mm
```

Paramètres de déclenchement des transferts

-tr

Facultatif. Indique une condition qui doit être vraie pour que ce transfert de fichier ait lieu. Si la condition n'est pas vérifiée, selon l'agent source, le transfert de fichiers est supprimé et aucun transfert n'a lieu. Indiquez le format suivant:

```
condition,namelist
```

où *condition* est l'une des valeurs suivantes:

file=existe

Il existe au moins un des fichiers de la liste de noms. En d'autres termes, si *l'un* des fichiers de la liste de noms existe, la condition est vraie.

fichier !=exist

Un minimum de l'un des fichiers de la liste de noms n'existe pas. En d'autres termes, si *l'un* des fichiers de la liste de noms n'existe pas, la condition est vraie.

filesize > =taille

Un minimum de l'un des fichiers de la liste de noms existe et sa taille minimale est spécifiée par *taille*. *size* est un entier avec une unité de taille facultative de Ko, Mo ou Go. Par exemple, `filesize ">" =10KB`. Si vous ne spécifiez pas d'unité de taille, la taille est supposée être en octets. Sur tous les systèmes d'exploitation, vous devez placer le symbole "supérieur à" (>) entre guillemets lorsque vous spécifiez l'option `filesize` sur la ligne de commande, comme illustré dans cet exemple.

Et où *namelist* est une liste séparée par des virgules de noms de fichiers situés sur le même système que l'agent source. Selon votre système d'exploitation, si vous souhaitez utiliser des noms de chemin ou des noms de fichier dans une liste de noms contenant des espaces, vous devrez peut-être les placer entre guillemets.

Vous pouvez spécifier plusieurs conditions de déclenchement en utilisant le paramètre **-tr** plusieurs fois. Toutefois, dans ce cas, chaque condition de déclenchement distincte doit être vraie pour que le transfert de fichiers ait lieu.

Remarque : Pour surveiller en permanence une ressource pour qu'une condition de déclenchement soit vraie, il est fortement recommandé d'utiliser la [surveillance des ressources](#). Vous pouvez créer un moniteur de ressources à l'aide de la commande `fteCreateMonitor`.

Dans l'exemple suivant, le fichier `file1.doc` est transféré de AGENT1 à AGENT2, à condition que le fichier `A.txt`, ou le fichier `B.txt`, ou les deux fichiers existent sur AGENT1 *et* que le fichier `A.txt`, ou le fichier `B.txt`, ou les deux fichiers soient égaux ou supérieurs à 1 Go:

```
fteCreateTransfer -sa AGENT1 -sm QM_JUPITER -da AGENT2 -dm QM_NEPTUNE
-tr file=exist,C:\export\A.txt,C:\export\B.txt
-tr filesize">"=1GB,C:\export\A.txt,C:\export\B.txt
-df C:\import\file1.doc C:\export\file1.doc
```

Vous pouvez combiner des paramètres de déclenchement avec des paramètres de planification. Si vous spécifiez les deux types de paramètres, les conditions de déclenchement sont appliquées au transfert de fichiers créé par les paramètres de planification.

Le paramètre **-tr** n'est pas pris en charge sur les agents de pont de protocole.

-tl

Facultatif. Indique si les échecs de déclenchement sont écrits dans le journal de transfert. Indiquez l'une des options suivantes :

oui

Des entrées de journal de transfert sont créées pour les transferts déclenchés ayant échoué. Il s'agit du comportement par défaut même si vous ne spécifiez pas le paramètre **-tl**.

non

Aucune entrée de journal de transfert n'est créée pour les transferts déclenchés ayant échoué.

Paramètres de spécification des options de transfert

-jn *nom_travail*

Facultatif. Identificateur de nom de travail défini par l'utilisateur qui est ajouté au message du journal de transfert lorsque le transfert démarre.

-md

Facultatif. Indique les métadonnées définies par l'utilisateur qui sont transmises aux points d'exit exécutés par l'agent. Le paramètre **-md** peut prendre une ou plusieurs paires nom-valeur séparées par des virgules. Chaque paire de noms se compose de *name=value*. Vous pouvez utiliser le paramètre **-md** plusieurs fois dans une commande.

Lorsque la propriété d'agent `enableUserMetadataOptions` est définie sur `true`, certaines clés de métadonnées définies par l'utilisateur fournissent des options supplémentaires pour le transfert. Pour plus d'informations sur les clés de métadonnées définies par l'utilisateur actuellement prises en charge, voir [«enableUserMetadataOptions: clés de métadonnées MFT définies par l'utilisateur prises en charge»](#), à la page 587. Lorsque la propriété `enableUserMetadataOptions` est définie sur `true`, les noms de clé commençant par `com.ibm.wmqfte.` ne sont pas pris en charge pour une utilisation définie par l'utilisateur.

-cs *somme de contrôle*

Facultatif. Indique si un algorithme de total de contrôle est exécuté sur les données de transfert de fichier pour vérifier l'intégrité des fichiers transférés. Indiquez l'une des options suivantes :

MD5

Calcule un total de contrôle MD5 pour les données. Le total de contrôle résultant pour les fichiers source et cible est consigné dans le journal de transfert à des fins de validation. Par défaut, Managed File Transfer calcule des totaux de contrôle MD5 pour tous les transferts de fichiers.

aucun

Aucun total de contrôle MD5 n'est calculé pour les données de transfert de fichier. Les enregistrements du journal de transfert dont le total de contrôle a été défini sur `aucun` et la valeur du total de contrôle est vide. Exemple :

```
<checksum method="none"></checksum>
```

Si vous utilisez l'option `none`, vous pouvez améliorer les performances de transfert de fichier, en fonction de votre environnement. Toutefois, la sélection de cette option signifie qu'il n'y a pas de validation des fichiers source ou de destination.

Si vous spécifiez le paramètre **-cs**, vous ne pouvez pas spécifier le paramètre **-td** car ces paramètres s'excluent mutuellement. Toutefois, vous pouvez spécifier un comportement de total de contrôle dans le fichier de définition de transfert.

-pr *priorité_transfert*

Facultatif. Indique le niveau de priorité du transfert. La priorité est une valeur comprise entre 0 et 9, où 0 est la priorité la plus basse. Le niveau de priorité par défaut est le niveau de priorité de l'agent source.

Cette valeur correspond à la valeur de priorité de message de IBM MQ. Pour plus d'informations, voir [Obtention de messages à partir d'une file d'attente: priorité](#). Le trafic des messages pour les données de transfert de fichiers est défini par défaut sur un niveau de priorité de 0, ce qui permet à votre trafic de messages IBM MQ d'être prioritaire.

-qmp booléen

Facultatif. Indique si les propriétés de message IBM MQ sont définies pour le premier message écrit dans la file d'attente de destination par le transfert. Les options valides sont les suivantes:

conforme

Définit les propriétés de message sur le premier message créé par le transfert.

false

Ne définit pas les propriétés de message sur le premier message créé par le transfert. Il s'agit de la valeur par défaut.

Vous ne pouvez spécifier le paramètre **-qmp** que si vous spécifiez également le paramètre **-dq**. Pour plus d'informations, voir «Propriétés de message MQ définies par MFT sur les messages écrits dans les files d'attente de destination», à la page 698

-qs taille_message

Facultatif. Indique si le fichier doit être fractionné en plusieurs messages de longueur fixe. Tous les messages ont le même ID groupe IBM MQ ; le dernier message du groupe a l'indicateur IBM MQ LAST_MSG_IN_GROUP défini. La taille des messages est spécifiée par la valeur de *message_size*. Le format de *message_size* est *longueurunités*, où *longueur* est un entier positif et *unités* est l'une des valeurs suivantes:

B

octets. La valeur minimale autorisée est égale à deux fois la valeur maximale en octets par caractère de la page de codes des messages de destination.

K

Cela équivaut à 1024 octets.

M

Cela équivaut à 1048576 octets.

Si le fichier est transféré en mode texte et se trouve dans un jeu de caractères codés sur deux octets ou dans un jeu de caractères codés sur plusieurs octets, il est scindé en messages sur la limite de caractères la plus proche de la taille de message spécifiée.

Vous ne pouvez spécifier le paramètre **-qs** que si vous spécifiez également le paramètre **-dq**. Vous ne pouvez spécifier qu'un seul des paramètres **-qs**, **-dqdbet** **-dqdt**.

-qi

Facultatif. L'utilisation de cette option inclut le délimiteur utilisé pour fractionner le fichier en plusieurs messages dans les messages. Le délimiteur est inclus au début ou à la fin du message, en fonction du paramètre **-dqdp** (qui spécifie le préfixe ou le suffixe). Par défaut, le délimiteur n'est pas inclus dans les messages.

Vous ne pouvez spécifier le paramètre **-qi** que si vous spécifiez également l'un des paramètres **-dqdt** et **-dqdb**.

-p options_configuration

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour créer le transfert de fichier. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination non défini par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de files d'attente de coordination par défaut est utilisé.

-w délai

Facultatif. Si vous spécifiez le paramètre **-w**, la commande **fteCreateTransfer** attend une réponse de l'agent avant de la renvoyer. Si vous ne spécifiez pas ce paramètre, la commande **fteCreateTransfer** attend au maximum cinq secondes pour recevoir un accusé de réception de l'agent source pour le transfert indiquant que l'agent a reçu la demande de transfert. Si aucun accusé

de réception n'est reçu pendant l'attente de cinq secondes, la commande **fteCreateTransfer** renvoie le message d'avertissement suivant:

```
BFGCL0253W: No acknowledgment to command from agent within timeout.
```

L'argument *timeout* est facultatif. Si vous spécifiez *timeout*, la commande **fteCreateTransfer** attend jusqu'à *timeout* secondes que l'agent réponde. Si l'agent ne répond pas avant que la limite de temps soit atteinte, la commande génère un avertissement et se termine par un code retour de 2. Si vous n'indiquez pas de valeur *timeout* ou si vous spécifiez une valeur *timeout* de -1, la commande attend que l'agent réponde.

V 9.0.1 **-rt délai_reprise**

Facultatif. Définit la durée, en secondes, pendant laquelle un agent source continue de tenter de récupérer un transfert de fichiers bloqué. Indiquez l'une des options suivantes :

-1

L'agent continue de tenter de reprendre le transfert bloqué jusqu'à ce que ce dernier soit terminé. Cette option est équivalente au comportement par défaut de l'agent si la propriété n'est pas définie.

0

L'agent arrête le transfert de fichiers dès qu'il entre dans la phase de reprise.

>0

L'agent continue de tenter de reprendre le transfert bloqué pendant le nombre de secondes défini par l'entier positif spécifié. Par exemple :

```
-rt 21600
```

indique que l'agent tente de récupérer le transfert pendant 6 heures à partir du moment où il entre en récupération. La valeur maximale de ce paramètre est 999999999.

La spécification de la valeur de délai d'attente de reprise de transfert de cette manière la définit par transfert. Pour définir une valeur globale pour tous les transferts dans un réseau Managed File Transfer, vous pouvez ajouter une propriété au [fichier agent.properties](#).

Paramètres d'appel des programmes

Pour plus d'informations sur la façon de démarrer un programme à partir de Managed File Transfer, voir «Spécification des programmes à exécuter avec MFT», à la page 256. Pour des exemples de spécification d'un programme à appeler à l'aide des paramètres décrits ici, voir [«Exemples d'utilisation de fteCreateTransfer pour démarrer des programmes»](#), à la page 869.

-presrc appel_source_préc

Facultatif. Indique un programme à appeler au niveau de l'agent source avant le démarrage du transfert. Utilisez le format suivant pour *pre_source_call*:

```
[type:]commandspec[, [retrycount][, [retrywait][, successsrc]]]
```

Dans cette syntaxe, les variables sont les suivantes:

type

Facultatif. Les valeurs valides sont **executable**, **antscriptet jcl**. La valeur par défaut est **executable**.

z/OS

La valeur **jcl** est applicable uniquement lorsqu'elle est ciblée sur un agent dans un environnement z/OS. Dans ce cas, la commande fait référence à un fichier ZFS, à un jeu de données lisible par QSAM ou à un membre d'un ensemble de données partitionnées. Le contenu doit être du JCL qui peut être soumis.

spéc. de commande

Obligatoire. La spécification de commande. Servez-vous de l'un des formats suivants :

- Entrez **executable**: `command[(arg1,arg2,...)]`

Si les arguments contiennent des substitutions de variables, telles que `${FilePath}` ou `${FileName}`, les variables sont remplacées par le premier élément de la demande de transfert.

Par exemple, si une demande de transfert se compose de fichiers "reports01.csv, reports02.csv, reports03.csv" et que le répertoire de destination est "/output", la demande de transfert suivante:

```
fteCreateTransfer -sa 1 -da 2 -presrc "executable:archive(${FileName})"
-dd TargetDir "${FilePath}" -gt task.xml
```

est remplacé par

```
fteCreateTransfer -sa 1 -da 2 -presrc "executable:archive(reports01.csv)"
-dd TargetDir "/ouptut" -gt task.xml
```

- Entrez **antscript**: `command[(name1=var1|target1,name2=var2|target2,...)]`
- Entrez **jcl**: `command`

où :

commande

Obligatoire. Nom du programme à appeler.

La valeur **jcl** est applicable uniquement lorsqu'elle est ciblée sur un agent dans un environnement z/OS .

Les arguments entre crochets ([]) sont facultatifs et la syntaxe dépend du type de commande. Les parenthèses, les virgules (,) et les barres obliques inversées (\) qui se trouvent dans la commande ou les paramètres doivent être échappées avec une barre oblique inverse (\).

Nombre de relances

Facultatif. Nombre de tentatives d'appel du programme si le programme ne renvoie pas de code retour. La valeur par défaut est 0.

attente de nouvelle tentative

Facultatif. Délai d'attente, en secondes, avant une nouvelle tentative d'appel du programme. La valeur par défaut est 0 (pas d'attente entre les nouvelles tentatives).

Code de réussite

Facultatif. Expression utilisée pour déterminer à quel moment l'appel de programme a abouti. Cette expression peut être composée d'une ou de plusieurs expressions. Combinez ces expressions avec une barre verticale (|) pour représenter la valeur booléenne OR ou une perluète (&) pour représenter la valeur booléenne AND. Chaque expression est de la forme suivante:

```
[>|<|!]value
```

Où

>

Facultatif. Test supérieur à de la *valeur*.

<

Facultatif. Test inférieur à de la *valeur*.

!

Facultatif. Un test différent de celui de la *valeur*.

valeur

Obligatoire. Un entier valide.

-predst *appel_pré_destination*

Facultatif. Indique un programme à appeler au niveau de l'agent de destination avant le début du transfert. *pre_destination_call* a le même format que *pre_source_call*.

-postsrc *appel_source_poste*

Facultatif. Indique un programme à appeler au niveau de l'agent source une fois le transfert terminé. *post_source_call* a le même format que *pre_source_call*.

-postdst *appel_destination_poste*

Facultatif. Indique un programme à appeler au niveau de l'agent de destination une fois le transfert terminé. *post_destination_call* a le même format que *pre_source_call*.

Paramètres de spécification de la destination

L'un des paramètres **-td**, **-df**, **-dd**, **-ds**, **-dqet** **-dp** est requis. Vous ne pouvez pas spécifier plusieurs de ces paramètres dans une demande de transfert ; ils s'excluent mutuellement.

-td *fichier_définition_transfert*

Facultatif. Nom du document XML qui définit une ou plusieurs spécifications de fichier source et de destination pour le transfert. Sinon, le nom du document XML qui contient une demande de transfert géré (qui peut avoir été générée par le paramètre **-gt**). Si vous spécifiez le paramètre **-td** et d'autres paramètres sur la ligne de commande, ces autres paramètres remplacent la valeur correspondante du fichier de définition de transfert.

La commande **fteCreateTransfer** localise le fichier de définition de transfert par rapport à votre répertoire en cours. Si vous ne pouvez pas utiliser la notation de chemin relatif pour spécifier l'emplacement du fichier de définition de transfert, utilisez le chemin d'accès complet et le nom de fichier du fichier de définition de transfert à la place.

z/OS Sous z/OS, vous devez stocker le fichier de définition de transfert dans un fichier UNIX sur z/OS UNIX System Services. Vous ne pouvez pas stocker les fichiers de définition de transfert dans des fichiers séquentiels z/OS ou des membres PDS.

IBM i Sous IBM i, vous devez stocker le fichier de définition de transfert dans le système de fichiers intégré.

Pour plus d'informations, voir [Utilisation des fichiers de définition de transfert](#).

-df *fichier_destination*

Facultatif. Nom du fichier de destination.

Si l'agent de destination est un agent de pont Connect:Direct, le fichier de destination est spécifié au format *connect_direct_node_name:file_path*. L'agent de pont Connect:Direct accepte uniquement les chemins de fichier spécifiés dans ce format. **z/OS** Si l'agent de destination est un agent de pont Connect:Direct et que la destination est un membre PDS, vous devez également spécifier le paramètre **-de** avec la valeur *overwrite*.

Entrez les informations suivantes :

- Si l'agent de destination est un agent de pont de protocole et que vous souhaitez spécifier un noeud final pour un fichier, utilisez le format suivant:

```
protocol_server:file_path
```

où *serveur_protocole* est le nom du serveur de protocole (facultatif) et où *chemin_fichier* est le chemin d'accès au fichier sur le système du serveur de protocole. Si vous ne spécifiez pas de serveur de protocole, le serveur de protocole par défaut est utilisé.

- Si vous souhaitez appeler l'un des exits utilisateur d'E-S de transfert Managed File Transfer que vous avez définis sur l'agent de destination, vous pouvez utiliser le paramètre **-df** dans un transfert.
-  Lorsque l'agent de destination est sous z/OS, si le fichier spécifié commence par //, il est supposé être un fichier z/OS partitionné.

-dd répertoire_destination

Facultatif. Nom du répertoire dans lequel le fichier est transféré. Indiquez un nom de répertoire valide sur le système sur lequel l'agent de destination s'exécute.

Si l'agent de destination est un agent de pont Connect:Direct, le répertoire de destination est spécifié au format *connect_direct_node_name:directory_path*. Si l'agent de destination est un agent de pont Connect:Direct et que la destination est un PDS, vous devez également spécifier le paramètre **-de** avec la valeur *overwrite*.

Entrez les informations suivantes :

- Si l'agent de destination est un agent de pont de protocole et que vous souhaitez spécifier un répertoire sur un noeud final particulier, utilisez le format suivant:

```
protocol_server:directory_path
```

où *serveur_protocole* est le nom du serveur de protocole (facultatif) et où *chemin_répertoire* est le chemin d'accès au répertoire sur le système du serveur de protocole. Si vous ne spécifiez pas de serveur de protocole, le serveur de protocole par défaut est utilisé.

- Si vous souhaitez appeler l'un des exits utilisateur d'E-S de transfert Managed File Transfer que vous avez définis sur l'agent de destination, vous pouvez utiliser le paramètre **-dd** dans un transfert.
-  Lorsque l'agent de destination est sous z/OS, si le fichier spécifié commence par //, il est supposé être un fichier partitionné z/OS.

-ds paramètre_données_séquentiel_destination

z/OS seulement. Facultatif. Nom du fichier séquentiel ou du membre PDS dans lequel les fichiers sont transférés. Indiquez un nom de fichier séquentiel ou un membre de fichier partitionné. Pour plus d'informations sur le transfert de fichiers, voir «Instructions pour le transfert de fichiers», à la page 648.

La syntaxe du nom de fichier est la suivante:

```
//data_set_name{;attribute(value);...;attribute(value)}
```

ou

```
//pds_data_set_name(member_name){;attribute(value);...;attribute(value)}
```

En d'autres termes, un spécificateur de nom de fichier préfixé avec // et éventuellement suivi d'un certain nombre d'attributs séparés par des points-virgules.

Par exemple :

```
//'TEST.FILE.NAME';DSNTYPE(PDS);RECFM(F,B);BLKSIZE(800);LRECL(80);CYL;SPACE(2,2)
```

Si le jeu de données se trouve sur un noeud Connect:Direct , vous devez préfixer le nom du jeu de données avec le nom du noeud. Exemple :

```
CD_NODE1:/' OBJECT.LIB' ;RECFM(F,B);BLKSIZE(800);LRECL(80)
```

Si l'agent de destination est un agent de pont Connect:Direct et que la destination est un membre PDS, vous devez également spécifier le paramètre **-de** avec la valeur `overwrite`. Pour plus d'informations sur les transferts de fichiers vers ou depuis des noeuds Connect:Direct , voir [«Transfert de fichiers vers et depuis des noeuds Connect:Direct»](#), à la page 653.

Pour les transferts impliquant uniquement des agents Managed File Transfer , si la partie nom de fichier est placée entre apostrophes, elle indique un nom de fichier qualifié complet. Si le nom de fichier n'est pas encadré par des guillemets simples, le système ajoute le qualificatif de haut niveau par défaut pour l'agent de destination (soit la valeur de la propriété d'agent `HLQ transferRoot`, soit l'ID utilisateur sous lequel l'agent s'exécute, si vous n'avez pas défini le qualificatif de haut niveau `transferRoot`).

Remarque :  Toutefois, pour les transferts impliquant un noeud Connect:Direct sur un système z/OS , la spécification de l'ensemble de données est interprétée comme un nom qualifié complet. Aucun qualificatif de haut niveau n'est ajouté au nom de fichier. C'est le cas même si le nom de fichier est encadré par des guillemets simples.

Lorsque vous transférez un fichier ou un ensemble de données sur bande, tout ensemble de données existant déjà sur la bande est remplacé. Les attributs du nouvel ensemble de données sont définis à partir des attributs transmis dans la définition de transfert. Si aucun attribut n'est spécifié, les attributs sont définis sur la même valeur que l'ensemble de données source ou sur les valeurs par défaut lorsque la source est un fichier. Les attributs d'un fichier bande existant sont ignorés.

Les attributs de fichier sont utilisés pour créer un fichier ou pour s'assurer qu'un fichier existant est compatible. La spécification des attributs de fichier est dans un format adapté à BPXWDYN (voir [Demande d'allocation dynamique](#) pour plus d'informations). Lorsque l'agent doit créer un fichier de destination, les attributs BPXWDYN suivants sont automatiquement spécifiés: *DSN* (*nom_fichier*) *NEW CATALOG MSG* (*numeric_file_descriptor*). La valeur de *numeric_file_descriptor* est générée par Managed File Transfer. Pour un transfert de fichier à fichier, les attributs de RECFM, LRECL et BLKSIZE de la source sont sélectionnés pour un nouveau fichier de destination. Le paramètre *SPACE* d'un nouveau fichier de destination n'est pas défini par Managed File Transfer et les valeurs par défaut du système sont utilisées. Par conséquent, il est recommandé de spécifier l'attribut *SPACE* lorsqu'un nouvel ensemble de données doit être créé. Vous pouvez utiliser la propriété **bpxwdynAllocAdditionalProperties** dans le fichier `agent.properties` pour définir les options BPXWDYN qui s'appliquent à tous les transferts. Pour plus d'informations, voir [«Le fichier MFT agent.properties»](#), à la page 564.

 Certaines options BPXWDYN ne doivent pas être spécifiées en cas d'utilisation de la commande **fteCreateTemplate**, de la commande **fteCreateTransfer** ou de la propriété **bpxwdynAllocAdditionalProperties** dans le fichier `agent.properties`. Pour obtenir la liste de ces propriétés, voir [«Propriétés BPXWDYN que vous ne devez pas utiliser avec MFT»](#), à la page 661.

Le paramètre **-ds** n'est pas pris en charge lorsque l'agent de destination est un agent de pont de protocole.

Si vous souhaitez appeler l'un des exits utilisateur d'E-S de transfert Managed File Transfer que vous avez définis sur un agent, n'indiquez pas le paramètre **-ds** dans un transfert. L'utilisation du paramètre **-ds** empêche les exits utilisateur d'E-S de transfert d'être appelés pour la destination et signifie que les E-S Managed File Transfer standard sont utilisées à la place.

 **-dp paramètre_données_partition_destination**

z/OS seulement. Facultatif. Nom de l'ensemble de données partitionnées de destination dans lequel les fichiers sont transférés. Indiquez un nom de fichier partitionné. Si un PDS est créé à la suite du transfert, ce PDS est créé en tant que PDSE par défaut. Vous pouvez remplacer la valeur par défaut en spécifiant `DSNTYPE=PDS`.

La syntaxe du nom de fichier PDS est la suivante:

```
//pds_data_set_name{;attribute;..;attribute}
```

La syntaxe du nom de fichier est la même que celle décrite pour le paramètre **-ds** (*destination_sequential_data_set*). Tous les détails de syntaxe pour la spécification des ensembles de données qui se trouvent sur les noeuds Connect:Direct s'appliquent également au paramètre **-dp**. Si l'agent de destination est un agent de pont Connect:Direct, vous devez également spécifier le paramètre **-de** avec la valeur *overwrite*.

Le paramètre **-dp** n'est pas pris en charge lorsque l'agent de destination est un agent de pont de protocole.

Si vous souhaitez appeler l'un des exits utilisateur d'E-S de transfert Managed File Transfer que vous avez définis sur un agent, n'indiquez pas le paramètre **-dp** dans un transfert. L'utilisation du paramètre **-dp** empêche les exits utilisateur d'E-S de transfert d'être appelés pour la destination et signifie que les E-S Managed File Transfer standard sont utilisées à la place.

-dq file_destination

Facultatif. Nom d'une file d'attente de destination vers laquelle les fichiers sont transférés. Vous pouvez éventuellement inclure un nom de gestionnaire de files d'attente dans cette spécification, en utilisant le format *QUEUE@QUEUEMANAGER*. Si vous n'indiquez pas de nom de gestionnaire de files d'attente, le nom du gestionnaire de files d'attente de l'agent de destination est utilisé. Vous devez indiquer un nom de file d'attente valide existant sur le gestionnaire de files d'attente.

Le paramètre **-dq** n'est pas pris en charge lorsque l'agent de destination est un agent de pont de protocole ou un agent de pont Connect:Direct, ou lorsque la spécification source est une file d'attente.

Si vous souhaitez appeler l'un des exits utilisateur d'E-S de transfert Managed File Transfer que vous avez définis sur un agent, n'indiquez pas le paramètre **-dq** dans un transfert. L'utilisation du paramètre **-dq** empêche les exits utilisateur d'E-S de transfert d'être appelés pour la destination et signifie que les E-S Managed File Transfer standard sont utilisées à la place.

-dqp persistent

Facultatif. Indique si les messages écrits dans la file d'attente de destination sont persistants. Les options valides sont les suivantes:

conforme

Écrit les messages persistants dans la file d'attente de destination. Il s'agit de la valeur par défaut.

false

Écrit les messages non persistants dans la file d'attente de destination.

Qdef

La valeur de persistance est extraite de l'attribut *DefPersistence* de la file d'attente de destination.

Vous ne pouvez spécifier le paramètre **-dqp** que si vous spécifiez également le paramètre **-dq**.

-dqdb délimiteur_hexadécimal

Facultatif. Indique le délimiteur hexadécimal à utiliser lors du fractionnement d'un fichier binaire en plusieurs messages. Tous les messages ont le même ID groupe IBM MQ; le dernier message du groupe a l'indicateur IBM MQ *LAST_MSG_IN_GROUP* défini. Le format de spécification d'un octet hexadécimal comme délimiteur est *xNN*, où *N* est un caractère compris entre 0-9 et a-f. Vous pouvez spécifier une séquence d'octets hexadécimaux comme délimiteur en spécifiant une liste d'octets hexadécimaux séparés par des virgules, par exemple: *x3e, x20, x20, xbf*.

Vous ne pouvez spécifier le paramètre **-dqdb** que si vous spécifiez également le paramètre **-dq** et que le transfert est en mode binaire. Vous ne pouvez spécifier qu'un seul des paramètres **-qs**, **-dqdbet** et **-dqdt**.

-dqdt modèle

Facultatif. Indique l'expression régulière Java à utiliser lors du fractionnement d'un fichier texte en plusieurs messages. Tous les messages ont le même ID groupe IBM MQ ; le dernier message du groupe a l'indicateur IBM MQ LAST_MSG_IN_GROUP défini. Le format de spécification d'une expression régulière comme délimiteur est une expression régulière placée entre parenthèses, (*regular_expression*) ou entre guillemets, "*regular_expression*". Pour plus d'informations, voir la section «Expressions régulières utilisées par MFT», à la page 679.

Par défaut, la longueur de la chaîne à laquelle l'expression régulière peut correspondre est limitée par l'agent de destination à cinq caractères. Vous pouvez modifier ce comportement en éditant la propriété d'agent **maxDelimiterMatchLength** . Pour plus d'informations, voir «Propriétés d'agent avancées», à la page 565.

Vous pouvez spécifier le paramètre **-dqdt** uniquement si vous spécifiez également le paramètre **-dq** et la valeur `text` pour le paramètre **-t** . Vous ne pouvez spécifier qu'un seul des paramètres **-qs**, **-dqdbet** **-dqdt** .

-dqdp position

Facultatif. Indique la position attendue du texte de destination et des délimiteurs binaires lors du fractionnement des fichiers. Vous ne pouvez spécifier le paramètre **-dqdp** que si vous spécifiez également l'un des paramètres **-dqdt** et **-dqdb** .

Indiquez l'une des options suivantes :

préfixe

Les délimiteurs sont attendus au début de chaque ligne.

suffixe

Les délimiteurs sont attendus à la fin de chaque ligne. Il s'agit de l'option par défaut.

-de comportement_fichier_destination

Facultatif. Indique l'action effectuée si un fichier de destination existe sur le système de destination. Les options valides sont les suivantes:

erreur

Signale une erreur et le fichier n'est pas transféré. Il s'agit de la valeur par défaut.

écrasement

Remplace le fichier cible existant.

Si vous spécifiez le paramètre **-de** , vous ne pouvez pas spécifier le paramètre **-td** car ces paramètres s'excluent mutuellement. Toutefois, vous pouvez spécifier le comportement du fichier de destination dans le fichier de définition de transfert.

-t type de transfert

Facultatif. Indique le type de transfert de fichier: mode binaire ou mode texte.

binaire

Les données du fichier sont transférées sans conversion. Il s'agit de la valeur par défaut.

texte

Les caractères de page de codes et de fin de ligne du fichier sont convertis. Vous pouvez spécifier la page de codes et la fin de ligne à utiliser pour la conversion avec les paramètres **-sce**, **-dce** ou **-dle** . Si vous ne spécifiez pas les paramètres **-sce**, **-dce** ou **-dle** , les conversions exactes effectuées dépendent du système d'exploitation de l'agent source et de l'agent cible.

 Par exemple, la page de codes d'un fichier transféré de Windows à z/OS est convertie de ASCII en EBCDIC. Lorsqu'un fichier est converti d'ASCII en EBCDIC, les caractères de fin de ligne sont convertis des paires de caractères de retour chariot (CR) et de saut de ligne (LF) ASCII en caractères de nouvelle ligne (NL) EBCDIC.

► **z/OS** Pour plus d'informations sur la façon dont les fichiers z/OS sont transférés, voir [Transfert de fichiers et de fichiers entre z/OS et des systèmes répartis](#) et [Transfert entre des fichiers](#).

Si vous spécifiez le paramètre **-t**, vous ne pouvez pas spécifier le paramètre **-td** car ces paramètres s'excluent mutuellement. Toutefois, vous pouvez spécifier le comportement du mode de transfert dans le fichier de définition de transfert.

-dce codage_caractère_destination

Facultatif. Indique le codage de caractères à utiliser pour écrire le fichier sur la destination. Cette option n'est applicable qu'aux fichiers texte. Par conséquent, **-t text** doit également être spécifié. Les pages de codes disponibles pour la conversion dépendent de la plateforme de la destination. Pour obtenir la liste des pages de codes disponibles, voir la rubrique «[Pages de codes disponibles pour MFT](#)», à la page 705.

noswaplfnl

Par défaut, Managed File Transfer utilise swaplfnl avec des jeux de caractères EBCDIC pris en charge. L'utilisation de swaplfnl modifie le comportement du mappage de jeu de caractères depuis et vers le caractère EBCDIC LF 0x25. Cependant, cela peut parfois entraîner un mappage qui n'est pas ce que vous souhaitez. Utilisez noswaplfnl pour remplacer ce comportement.

-dle fin_ligne_destination

Facultatif. Indique les caractères de fin de ligne utilisés lorsque le fichier est écrit à la destination. Cette option s'appliquant uniquement aux fichiers texte, vous devez également spécifier le paramètre **-t text**. Les options valides sont :

Saut de ligne

Saut de ligne. Il s'agit de la valeur par défaut pour les plateformes suivantes:

- ► **UNIX** UNIX
- ► **z/OS** Fichiers z/OS UNIX System Services

Lorsque vous utilisez les pages de codes EBCDIC standard fournies avec Managed File Transfer pour les fichiers EBCDIC, les caractères de fin de ligne sont mappés à un caractère NL (0x15) et non à un caractère LF (0x25).

CRLF

Retour chariot suivi d'un saut de ligne. ► **Windows** Il s'agit de la valeur par défaut pour Windows.

► **z/OS** Si la destination du transfert est un fichier z/OS, cette option est ignorée.

-dtr

Facultatif. Indique que les enregistrements de destination dont la longueur est supérieure à l'attribut de fichier LRECL sont tronqués. Si ce paramètre n'est pas spécifié, les enregistrements sont encapsulés. Ce paramètre est valide uniquement pour les transferts en mode texte où la destination est un fichier.

-dfa attributs

Facultatif. Lors d'un transfert vers un agent IBM MQ 8.0 Managed File Transfer s'exécutant sur un système 4690, ce paramètre est utilisé pour spécifier une liste d'attributs de fichier séparés par des points-virgules qui sont associés aux fichiers de destination dans le transfert. Le paramètre **-dfa** peut être spécifié avec ou sans valeur. Par exemple, sans valeur:

```
-dfa ATTRIBUTE1;ATTRIBUTE2
```

Par exemple, avec une valeur:

```
-dfa ATTRIBUTE1(VALUE);ATTRIBUTE2(VALUE)
```

Par exemple, un attribut avec une valeur et un autre sans:

```
-dfa ATTRIBUTE1;ATTRIBUTE2(VALUE)
```

Vous pouvez utiliser le paramètre **-dfa** plusieurs fois dans une commande.

Pour plus d'informations sur les attributs de fichier sur 4690, voir [Attributs de distribution de fichier](#) dans IBM Documentation.

Paramètres de sécurité

-mquserid (userID)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de files d'attente de commandes.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de commandes. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

Paramètres de spécification de la source

-sd disposition-fichier-source

Facultatif. Indique l'action effectuée sur un fichier source dans les transferts de fichier à fichier ou de fichier à message lorsque ce fichier source est transféré vers sa destination. Les options valides sont les suivantes:

Quitter

Les fichiers source restent inchangés. Il s'agit de la valeur par défaut.

supprimer

Les fichiers source sont supprimés du système source une fois que les fichiers source ont été transférés.

Remarque : Pour les transferts de message à fichier, les messages de la file d'attente source sont toujours supprimés une fois qu'ils ont été transférés. Cela signifie que si le paramètre **-sd** est défini sur `leave` pour un transfert de message vers fichier, la valeur est ignorée.

 Sous z/OS, si la source est un fichier de bande et que vous spécifiez l'option `delete`, la bande est remontée pour supprimer le fichier. Ce comportement est dû au comportement de l'environnement système.

Si la source est une file d'attente et que vous spécifiez l'option `leave`, la commande renvoie une erreur et un transfert n'est pas demandé.

Si l'agent source est un agent de pont Connect:Direct et que vous spécifiez l'option `delete`, le comportement est différent du comportement d'élimination de la source habituel. L'un des cas suivants se produit:

- Si Connect:Direct utilise un processus généré par Managed File Transfer pour déplacer le fichier ou l'ensemble de données depuis la source, l'indication de l'option `delete` entraîne l'échec du transfert. Pour spécifier que le fichier source doit être supprimé, vous devez soumettre un processus Connect:Direct défini par l'utilisateur. Pour plus d'informations, voir la section [«Soumission d'un processus Connect:Direct défini par l'utilisateur à partir d'une demande de transfert de fichier»](#), à la page 245.
- Si Connect:Direct fait appel à un processus défini par l'utilisateur pour déplacer le fichier ou l'ensemble de données à partir de la source, ce paramètre est transmis au processus via la variable symbolique intrinsèque **%FTEFDISP**. Le processus défini par l'utilisateur détermine si la source est supprimé. Le résultat du transfert dépend du résultat renvoyé par le processus défini par l'utilisateur.

Si vous spécifiez le paramètre **-sd**, vous ne pouvez pas spécifier le paramètre **-td** car ces paramètres s'excluent mutuellement. Toutefois, vous pouvez spécifier le comportement de disposition de la source dans le fichier de définition de transfert.

-r

Facultatif. Transfert récursif des fichiers dans les sous-répertoires lorsque *source_specification* contient des caractères génériques. Lorsque Managed File Transfer est présenté avec un caractère générique comme *source_specification*, tous les sous-répertoires qui correspondent au caractère générique sont transférés uniquement si vous spécifiez le paramètre **-r**. Lorsque *source_specification* correspond à un sous-répertoire, tous les fichiers de ce répertoire et de ses sous-répertoires (y compris les fichiers masqués) sont toujours transférés.

Pour plus d'informations sur la façon dont Managed File Transfer gère les caractères génériques, voir [Utilisation de caractères génériques](#)

Si vous spécifiez le paramètre **-r**, vous ne pouvez pas spécifier le paramètre **-td** car ces paramètres s'excluent mutuellement. Toutefois, vous pouvez spécifier un comportement récursif dans le fichier de définition de transfert.

-scc codage_caractères_source

Facultatif. Indique le codage de caractères à utiliser pour lire le fichier source lors de la conversion de caractères. Cette option n'est applicable qu'aux fichiers texte. Par conséquent, **-t text** doit également être spécifié. Les pages de codes disponibles pour la conversion dépendent de la plateforme de l'agent de destination, car la conversion est effectuée sur le système de destination. Pour obtenir la liste des pages de codes disponibles, voir la rubrique [«Pages de codes disponibles pour MFT»](#), à la page 705.

noswaplfnl

Par défaut, Managed File Transfer utilise swaplfnl avec des jeux de caractères EBCDIC pris en charge. L'utilisation de swaplfnl modifie le comportement du mappage de jeu de caractères depuis et vers le caractère EBCDIC LF 0x25. Cependant, cela peut parfois entraîner un mappage qui n'est pas ce que vous souhaitez. Utilisez noswaplfnl pour remplacer ce comportement.

z/OS

-skeep

Facultatif. Indique que les espaces de fin sont conservés sur les enregistrements source lus à partir d'un fichier orienté enregistrement de longueur fixe (par exemple, un fichier z/OS) dans le cadre d'un transfert en mode texte. Si vous ne spécifiez pas ce paramètre, les espaces de fin sont supprimés des enregistrements source.

z/OS

-srdb délimiteur

Facultatif. Pour les fichiers source orientés enregistrement (par exemple, les fichiers z/OS), indique une ou plusieurs valeurs d'octet à insérer comme délimiteur lors de l'ajout d'enregistrements dans un fichier binaire. Vous devez spécifier chaque valeur sous la forme de deux chiffres hexadécimaux dans la plage 00-FF, précédés de x. Séparez les octets par des virgules. Exemple :

```
-srdb x0A
```

ou

```
-srdb x0D,x0A
```

Vous devez configurer le transfert en mode binaire.

z/OS

-srdp position

Facultatif. Indique la position d'insertion des délimiteurs d'enregistrement source. Vous ne pouvez spécifier le paramètre **-srdp** que si vous spécifiez également le paramètre **-srdb**.

Indiquez l'une des options suivantes :

préfixe

Les délimiteurs sont insérés au début de chaque enregistrement.

suffixe

Les délimiteurs sont insérés à la fin de chaque enregistrement. Il s'agit de l'option par défaut.

-sq

Facultatif. Indique que la source d'un transfert est une file d'attente.

Si vous souhaitez appeler l'un des exits utilisateur d'E-S de transfert Managed File Transfer que vous avez définis sur un agent, n'indiquez pas le paramètre **-sq** dans un transfert. L'utilisation du paramètre **-sq** empêche les exits utilisateur d'E-S de transfert d'être appelés pour la source et signifie que l'entrée-sortie Managed File Transfer standard est utilisée à la place.

-sqgi

Facultatif. Indique que les messages sont regroupés par ID groupe IBM MQ . Le premier groupe complet est écrit dans le fichier de destination. Si ce paramètre n'est pas spécifié, tous les messages de la file d'attente source sont écrits dans le fichier de destination.

Vous ne pouvez spécifier le paramètre **-sqgi** que si vous spécifiez également le paramètre **-sq** .

-sqdt délimiteur de texte

Facultatif. Indique une séquence de texte à insérer comme délimiteur lors de l'ajout de plusieurs messages à un fichier texte. Vous pouvez inclure des séquences d'échappement Java pour les littéraux chaîne dans le délimiteur. Par exemple, `-sqdt \u007d\n`.

Le délimiteur de texte est codé au format binaire à l'aide du codage source du transfert. Chaque message est lu au format binaire. Le délimiteur codé est ajouté en préfixe ou en suffixe au format binaire au message (comme indiqué par le paramètre **-sqdp**) et le résultat est transféré au format binaire à l'agent de destination. Si la page de codes de l'agent source inclut des états d'arrivée et de sortie, l'agent suppose que chaque message est à l'état de sortie à la fin du message. Au niveau de l'agent de destination, les données binaires sont converties de la même manière qu'un transfert de texte de fichier à fichier.

Vous pouvez spécifier le paramètre **-sqdt** uniquement si vous spécifiez également le paramètre **-sq** et la valeur `text` pour le paramètre **-t** .

-sqdb délimiteur_hexadécimal

Facultatif. Indique une ou plusieurs valeurs d'octet à insérer comme délimiteur lors de l'ajout de plusieurs messages à un fichier binaire. Chaque valeur doit être spécifiée sous la forme de deux chiffres hexadécimaux dans la plage 00-FF, précédés de x. Plusieurs octets doivent être séparés par des virgules. Par exemple, `-sqdb x08,xA4`.

Vous ne pouvez spécifier le paramètre **-sqdb** que si vous spécifiez également le paramètre **-sq** . Vous ne pouvez pas spécifier le paramètre **-sqdb** si vous spécifiez également la valeur `text` pour le paramètre **-t** .

-sqdp position

Facultatif. Indique la position d'insertion du texte source et des délimiteurs binaires. Vous ne pouvez spécifier le paramètre **-sqdp** que si vous avez également spécifié l'un des paramètres **-sqdt** et **-sqdb** .

Indiquez l'une des options suivantes :

préfixe

Les délimiteurs sont insérés au début de chaque message

suffixe

Les délimiteurs sont insérés à la fin de chaque message. Il s'agit de l'option par défaut.

-sqwt temps_attente

Facultatif. Indique le délai d'attente, en secondes, pour que l'une des conditions suivantes soit remplie:

- Pour qu'un nouveau message apparaisse dans la file d'attente
- Si le paramètre **-sqgi** a été spécifié, pour qu'un groupe complet apparaisse dans la file d'attente

Si aucune de ces conditions n'est remplie dans le délai spécifié par *wait_time*, l'agent source arrête la lecture à partir de la file d'attente et termine le transfert. Si le paramètre **-sqwt** n'est pas spécifié, l'agent source arrête immédiatement la lecture à partir de la file d'attente source si la file d'attente source est vide ou, dans le cas où le paramètre **-sqgi** est spécifié, s'il n'y a pas de groupe complet dans la file d'attente.

Pour plus d'informations sur l'utilisation du paramètre **-sqwt**, voir «Conseils pour la spécification d'un temps d'attente sur un transfert de message vers fichier», à la page 704.

Vous ne pouvez spécifier le paramètre **-sqwt** que si vous spécifiez également le paramètre **-sq**.

spécif_source

Une ou plusieurs spécifications de fichier qui déterminent la ou les sources du transfert de fichiers.

Obligatoire si vous spécifiez l'un des paramètres **-df**, **-dd**, **-dp**, **-dq** ou **-ds**. Si vous spécifiez le paramètre **-td**, ne spécifiez pas *source_specification*.

- Si vous n'avez pas spécifié le paramètre **-sq**, *source_specification* est une ou plusieurs spécifications de fichier qui déterminent la ou les sources du transfert de fichier. Les spécifications de fichier peuvent prendre l'une des cinq formes et inclure des caractères génériques. Pour plus d'informations sur les caractères génériques, voir «Utilisation de caractères génériques avec MFT», à la page 673. Vous pouvez mettre en échappement les astérisques qui font partie de la spécification de fichier en utilisant deux astérisques (**) dans la spécification de fichier.

Vous pouvez spécifier plusieurs spécifications de fichier source séparées par un espace. Toutefois, si vous spécifiez plusieurs spécifications source pour les paramètres **-df** ou **-ds** et que vous spécifiez également **-de overwrite**, la destination contiendra uniquement les données du fichier source que vous avez spécifié en dernier. Si vous ne spécifiez pas **-de overwrite**, le transfert ne peut aboutir que partiellement. Si le fichier de destination n'existait pas auparavant, il contiendra les données du fichier source que vous avez indiqué en premier.

Pour transférer des fichiers qui contiennent des espaces dans leurs noms de fichier, par exemple a b.txt vers le fichier c d.txt, placez les noms de fichier qui contiennent des espaces entre guillemets. Spécifiez le texte suivant dans le cadre de la commande **fteCreateTransfer** :

```
-df "c d.txt" "a b.txt"
```

Chaque spécification de fichier doit appartenir à l'une des catégories suivantes:

Noms de fichiers

Nom d'un fichier, exprimé dans la notation appropriée pour le système sur lequel l'agent source s'exécute. Lorsqu'un nom de fichier est spécifié en tant que spécification de fichier source, le contenu du fichier est copié.

Répertoires

Nom d'un répertoire, exprimé dans la notation appropriée pour le système sur lequel l'agent source est en cours d'exécution. Lorsqu'un répertoire est spécifié en tant que spécification de fichier source, le contenu du répertoire est copié. Plus précisément, tous les fichiers du répertoire et de tous ses sous-répertoires, y compris les fichiers masqués, sont copiés.

Par exemple, pour copier le contenu de DIR1 dans DIR2 uniquement, indiquez `fteCreateTransfer ... -dd DIR2 DIR1/*`

z/OS Fichier séquentiel

Nom d'un fichier séquentiel ou d'un membre de fichier partitionné. Dénoter les ensembles de données en faisant précéder le nom de l'ensemble de données de deux barres obliques (/).

Si vous spécifiez un agent de pont de protocole comme agent source, vous ne pouvez pas spécifier un ensemble de données comme spécification de fichier source.

Fichier partitionné

Nom d'un fichier partitionné. Dénoter les noms de fichier en les faisant précéder de deux barres obliques (/).

Si vous spécifiez un agent de pont de protocole comme agent source, vous ne pouvez pas spécifier un ensemble de données comme spécification de fichier source.

Nom de fichier ou répertoire sur un noeud Connect:Direct

(agent de pontConnect:Direct uniquement). Nom d'un noeud Connect:Direct , un signe deux-points (:) et un chemin de fichier ou de répertoire sur le système qui héberge le noeud Connect:Direct . Par exemple, *connect_direct_node_name:file_path*.

Si l'agent source est un agent de pont Connect:Direct , il n'accepte que les spécifications de source sous cette forme.

Remarque : Les caractères génériques ne sont pas pris en charge dans les chemins de fichier lorsque l'agent source est un agent de pont Connect:Direct .

Nom de fichier ou répertoire sur un serveur de fichiers de protocole

Nom d'un serveur de fichiers de protocole, un signe deux-points (:) et un chemin de fichier ou de répertoire sur le système du serveur de protocole. Par exemple, *protocol_server:file_path*.

Si vous ne spécifiez pas de serveur de protocole, le serveur de protocole par défaut est utilisé.

- Si vous spécifiez le paramètre **-sq** , *source_specification* est le nom d'une file d'attente locale sur le gestionnaire de files d'attente de l'agent source. Vous ne pouvez spécifier qu'une seule file d'attente source. La file d'attente source est spécifiée au format suivant:

```
QUEUE_NAME
```

Le nom du gestionnaire de files d'attente n'est pas inclus dans la spécification de file d'attente source car le gestionnaire de files d'attente doit être identique au gestionnaire de files d'attente de l'agent source.

-  Si l'agent source est sous z/OS, les fichiers source qui commencent par // sont supposés être des fichiers partitionnés z/OS .

Autres paramètres

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemples

Dans cet exemple de base, le fichier *originalfile.txt* est transféré de AGENT1 à AGENT2 sur le même système et renommé en *transferredfile.txt*

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -df C:\import\transferredfile.txt C:\export\originalfile.txt
```

Dans cet exemple, les fichiers *originalfile.txt* et *originalfile2.txt* sont transférés de AGENT1 à AGENT2 sur le même système, dans le répertoire *C:\import*

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -dd C:\import C:\export\originalfile.txt  
C:\export\originalfile2.txt
```

Dans cet exemple, le fichier `originalfile.txt` est transféré du système AGENT1 vers le système AGENT2. Le transfert de fichiers est planifié pour avoir lieu à 09:00 en fonction de l'heure système du système de l'agent source et se produit toutes les deux heures quatre fois:

```
fteCreateTransfer -sa AGENT1 -sm QM_JUPITER -da AGENT2 -dm QM_NEPTUNE
-tb source -ss 09:00 -oi hours -of 2 -oc 4
-df C:\import\transferredfile.txt C:\export\originalfile.txt
```

Dans cet exemple, le fichier `originalfile.txt` est transféré de AGENT1 à AGENT2, à condition que le fichier `A.txt` existe sur AGENT1:

```
fteCreateTransfer -sa AGENT1 -sm QM_JUPITER -da AGENT2 -dm QM_NEPTUNE
-tr file=exist,C:\export\A.txt -df C:\import\transferredfile.txt C:\export\originalfile.txt
```

z/OS Dans cet exemple, le fichier `originalfile.txt` est transféré du système AGENT1 vers un ensemble de données `/'USERID.TRANS.FILE.TXT'` sur le système AGENT2. Le mode texte est sélectionné pour convertir les données d'ASCII en EBCDIC.

```
fteCreateTransfer -t text -sa AGENT1 -da AGENT2
-ds "/'TRANS.FILE.TXT;RECFM(V,B);BLKSIZE(6144);LRECL(1028);
SPACE(5,1)" C:\export\originalfile.txt
```

z/OS Dans cet exemple, un membre d'un fichier qualifié complet sur le système AGENT1 est transféré dans un fichier sur le système AGENT2. Le mode texte est sélectionné pour convertir le fichier EBCDIC en page de codes par défaut du système AGENT2.

```
fteCreateTransfer -t text -sa AGENT1 -da AGENT2 -df /tmp/IEEUJV.txt "/'SYS1.SAMPLIB(IEEUJV)'"
```

Dans cet exemple, un fichier appelé `file.bin` sur l'agent AGENT1 est transféré vers un fichier de destination appelé `file.bin` sur le serveur de fichiers de protocole `accountshost.ibm.com` à l'aide de l'agent de destination `BRIDGE1`.

```
fteCreateTransfer -sa AGENT1 -da BRIDGE1 -df accountshost.ibm.com:/tmp/file.bin /tmp/file.bin
```

Dans cet exemple, un caractère générique est utilisé sans guillemets. Tous les fichiers du répertoire de travail en cours de AGENT1 qui se terminent par `.txt` sont transférés dans le répertoire `C:\import` sur AGENT2. Les noms de fichier restent inchangés.

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -dd C:\import *.txt
```

Dans cet exemple, un caractère générique est utilisé avec des guillemets. Tous les fichiers du répertoire racine de transfert de AGENT1 qui se terminent par `.txt` sont transférés dans le répertoire `C:\import` sur AGENT2. Les noms de fichier restent inchangés.

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -dd C:\import "*.txt"
```

Codes retour

Code retour	Description
0	La commande a abouti.
1	La commande a échoué.

Code retour	Description
2	La commande s'est terminée avec un délai d'attente. La commande a envoyé un message à l'agent, mais celui-ci n'a pas répondu dans le délai spécifié.
20	La commande s'est terminée avec un succès partiel et certains fichiers ont été transférés.
21	Le gestionnaire de files d'attente auquel la commande fteCreateTransfer était connectée a été arrêté avant que le résultat du transfert ne soit déterminé.
40	Echec. Aucun des fichiers indiqués n'a été transféré.
41	Le transfert a été annulé.
42	Le transfert n'a pas eu lieu parce que le transfert était conditionnel et que la condition requise n'était pas remplie.
43	Le message de demande de transfert a été syntaxiquement incorrect.
44	L'agent source ne disposait pas de la capacité suffisante pour effectuer le transfert.
45	L'agent de destination ne disposait pas de la capacité suffisante pour effectuer le transfert.
46	Le nombre de fichiers transférés a dépassé la limite de l'agent source.
47	Le nombre de fichiers transférés dépasse la limite de l'agent cible.

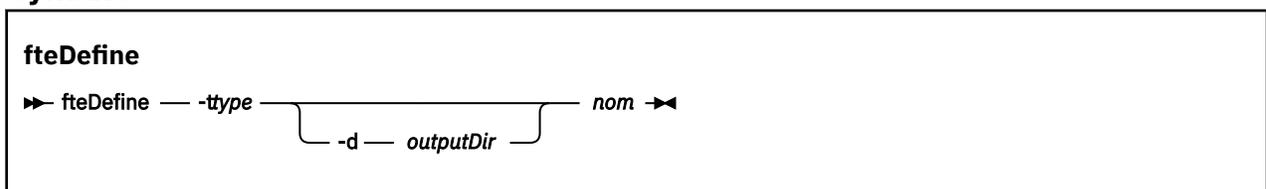
fteDefine (génération de scripts de configuration)

La commande **fteDefine** permet de générer les scripts de configuration nécessaires pour définir les objets Gestionnaire de files d'attente d'agent spécifiés.

Objet

Vous utilisez la commande **fteDefine** lorsque certaines étapes de configuration doivent être exécutées sur un système distant par rapport à celui comportant les données de configuration, par exemple si vous configurez les files d'attente pour un agent dans un gestionnaire de files d'attente accessible via une connexion client.

Syntaxe



Paramètres

-t type

Obligatoire. Type d'objet à définir. Les options pour le type sont agent.

-d rép_sortie

Facultatif. Chemin de répertoire dans lequel les scripts sont écrits. S'il n'est pas indiqué, les scripts sont écrits dans le flux de sortie standard.

nom

Obligatoire. Un ou plusieurs noms d'objet à définir. Si vous spécifiez plusieurs noms d'objet, séparez-les par un espace. Exemple: *name1 name2 . . .*

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemples

Dans cet exemple, la commande **fteDefine** est spécifiée avec le paramètre **-t agent** et un nom d'agent unique. La sortie est écrite dans un fichier.

```
fteDefine -t agent EXAMPLE.AGENT >EXAMPLE.AGENT_create.mqsc
```

La sortie qui est générée depuis cette commande correspond aux scripts de commandes MQSC à exécuter dans le gestionnaire de file d'attente d'agent afin de créer les files d'attente d'agent nécessaires :

```
$ fteDefine -t agent EXAMPLE.AGENT
5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
DEFINE QLOCAL(SYSTEM.FTE.COMMAND.EXAMPLE.AGENT) +
  DEFPRTY(0) +
  DEFSOPT(SHARED) +
  GET(ENABLED) +
  MAXDEPTH(5000) +
  MAXMSGL(4194304) +
  MSGDLVSQ(PRIORITY) +
  PUT(ENABLED) +
  RETINTVL(99999999) +
  SHARE +
  NOTRIGGER +
  USAGE(NORMAL) +
  REPLACE
DEFINE QLOCAL(SYSTEM.FTE.DATA.EXAMPLE.AGENT) +
  DEFPRTY(0) +
  DEFSOPT(SHARED) +
  GET(ENABLED) +
  MAXDEPTH(5000) +
  MAXMSGL(4194304) +
  MSGDLVSQ(PRIORITY) +
  PUT(ENABLED) +
  RETINTVL(99999999) +
  SHARE +
  NOTRIGGER +
  USAGE(NORMAL) +
  REPLACE
...
etc.
```

Dans cet exemple, la commande **fteDefine** est spécifiée avec le paramètre **-d outputDir** et plusieurs noms d'agent.

```
fteDefine -t agent -d /tmp EXAMPLE.AGENT.1 EXAMPLE.AGENT.2 EXAMPLE.AGENT.3
```

La sortie qui est générée par cette commande contient les chemins d'accès absolus aux fichiers de script de commandes MQSC :

```
$ fteDefine -t agent -d /tmp EXAMPLE.AGENT.1 EXAMPLE.AGENT.2 EXAMPLE.AGENT.3
5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
BFGCM0239I: A file has been created containing the MQSC definitions to define the agent
EXAMPLE.AGENT.1.
The file can be found here: '/tmp/EXAMPLE.AGENT.1_create.mqsc'.
BFGCM0239I: A file has been created containing the MQSC definitions to define the agent
EXAMPLE.AGENT.2.
The file can be found here: '/tmp/EXAMPLE.AGENT.2_create.mqsc'.
BFGCM0239I: A file has been created containing the MQSC definitions to define the agent
EXAMPLE.AGENT.3.
The file can be found here: '/tmp/EXAMPLE.AGENT.3_create.mqsc'.
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Référence associée

«fteDelete (générer des scripts pour supprimer la configuration)», à la page 472

La commande **fteDelete** permet de générer les scripts de configuration nécessaires pour supprimer les objets Gestionnaire de files d'attente d'agent spécifiés.

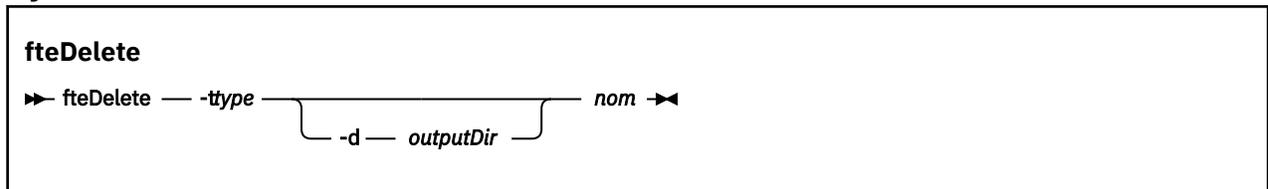
fteDelete (générer des scripts pour supprimer la configuration)

La commande **fteDelete** permet de générer les scripts de configuration nécessaires pour supprimer les objets Gestionnaire de files d'attente d'agent spécifiés.

Objet

Vous devez utiliser la commande **fteDelete** lorsque certaines étapes de configuration doivent être exécutées sur un système distant de celui qui contient les données de configuration. Par exemple, la suppression des files d'attente d'un agent client distant sur un gestionnaire de files d'attente local.

Syntaxe



Paramètres

-t *type*

Obligatoire. Type d'objet à supprimer. Les options pour le type sont agent.

-d *rép_sortie*

Facultatif. Chemin de répertoire dans lequel les scripts sont écrits. S'il n'est pas indiqué, les scripts sont écrits dans le flux de sortie standard.

nom

Obligatoire. Un ou plusieurs noms des objets à supprimer. Si vous spécifiez plusieurs noms d'objet, séparez-les par un espace. Exemple: *name1 name2 . . .*

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemples

Dans cet exemple, la commande **fteDelete** est spécifiée avec le paramètre **-t agent** et un nom d'agent unique. La sortie est écrite dans un fichier.

```
fteDelete -t agent EXAMPLE.AGENT >EXAMPLE.AGENT_delete.mqsc
```

La sortie générée à partir de cette commande est les scripts de commande MQSC à exécuter sur le gestionnaire de files d'attente d'agent pour supprimer les files d'attente d'agent:

```
$ fteDelete -t agent EXAMPLE.AGENT
5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
CLEAR QLOCAL(SYSTEM.FTE.COMMAND.EXAMPLE.AGENT)
DELETE QLOCAL(SYSTEM.FTE.COMMAND.EXAMPLE.AGENT)
CLEAR QLOCAL(SYSTEM.FTE.DATA.EXAMPLE.AGENT)
DELETE QLOCAL(SYSTEM.FTE.DATA.EXAMPLE.AGENT)
CLEAR QLOCAL(SYSTEM.FTE.REPLY.EXAMPLE.AGENT)
DELETE QLOCAL(SYSTEM.FTE.REPLY.EXAMPLE.AGENT)
CLEAR QLOCAL(SYSTEM.FTE.STATE.EXAMPLE.AGENT)
DELETE QLOCAL(SYSTEM.FTE.STATE.EXAMPLE.AGENT)
CLEAR QLOCAL(SYSTEM.FTE.EVENT.EXAMPLE.AGENT)
DELETE QLOCAL(SYSTEM.FTE.EVENT.EXAMPLE.AGENT)
```

...
etc.

Dans cet exemple, la commande **fteDelete** est spécifiée avec le paramètre **-d outputDir** et plusieurs noms d'agent.

```
fteDelete -t agent -d /tmp EXAMPLE.AGENT.1 EXAMPLE.AGENT.2 EXAMPLE.AGENT.3
```

La sortie qui est générée par cette commande contient les chemins d'accès absolus aux fichiers de script de commandes MQSC :

```
$ fteDelete -t agent -d /tmp EXAMPLE.AGENT.1 EXAMPLE.AGENT.2 EXAMPLE.AGENT.3
5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
BFGCM0241I: A file has been created containing the MQSC definitions to delete the agent
EXAMPLE.AGENT.1.
The file can be found here: '/tmp/EXAMPLE.AGENT.1_delete.mqsc'.
BFGCM0241I: A file has been created containing the MQSC definitions to delete the agent
EXAMPLE.AGENT.2.
The file can be found here: '/tmp/EXAMPLE.AGENT.2_delete.mqsc'.
BFGCM0241I: A file has been created containing the MQSC definitions to delete the agent
EXAMPLE.AGENT.3.
The file can be found here: '/tmp/EXAMPLE.AGENT.3_delete.mqsc'.
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Référence associée

«[fteDefine \(génération de scripts de configuration\)](#)», à la page 470

La commande **fteDefine** permet de générer les scripts de configuration nécessaires pour définir les objets Gestionnaire de files d'attente d'agent spécifiés.

fteDeleteAgent

La commande **fteDeleteAgent** supprime un Managed File Transfer Agent et sa configuration. Si l'agent est un agent de pont de protocole, le fichier de données d'identification de l'utilisateur est laissé sur le système de fichiers.

Objet

Arrêtez l'agent à l'aide de la commande [fteStopAgent](#) avant d'exécuter la commande **fteDeleteAgent**.

Si vous avez configuré votre agent pour qu'il s'exécute en tant que service Windows, l'exécution de la commande **fteDeleteAgent** supprime la définition de service.

V 9.0.4 Depuis la IBM MQ 9.0.4, tout moniteur de ressources et tout transfert planifié sont supprimés lorsque l'agent est supprimé.

Seuls les utilisateurs qui sont des administrateurs IBM MQ (et des membres du groupe mqm) peuvent exécuter cette commande. Si vous tentez d'exécuter cette commande en tant qu'utilisateur qui n'est pas un administrateur IBM MQ, vous recevrez un message d'erreur et la commande ne sera pas exécutée.

La commande **fteDeleteAgent** fournit les commandes MQSC que vous devez exécuter sur le gestionnaire de files d'attente de votre agent pour effacer et supprimer les files d'attente système de l'agent. Ces files d'attente sont les suivantes:

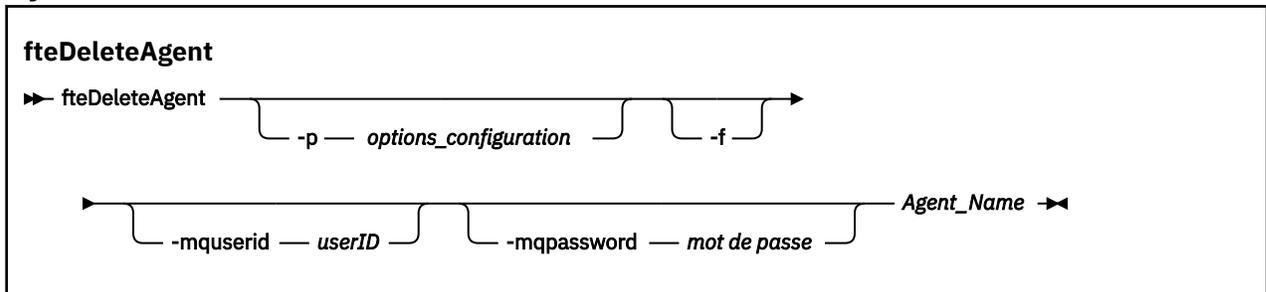
- SYSTEM.FTE.AUTHADM1.*nom_agent*
- SYSTEM.FTE.AUTHAGT1.*nom_agent*

- SYSTEM.FTE.AUTHMON1.*nom_agent*
- SYSTEM.FTE.AUTHOPS1.*nom_agent*
- SYSTEM.FTE.AUTHSCH1.*nom_agent*
- SYSTEM.FTE.AUTHTRN1.*nom_agent*
- 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*

La commande **fteCreateAgent** fournit également ces commandes dans un fichier à l'emplacement suivant:

`MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name/agent_name_delete.mqsc`

Syntaxe



Paramètres

-p (*options_configuration*)

Facultatif. Si vous disposez de plusieurs gestionnaires de files d'attente de coordination, utilisez ce paramètre pour spécifier explicitement la configuration d'agent à supprimer. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite les options de configuration associées à ce gestionnaire de files d'attente de coordination non défini par défaut.

Spécifiez le paramètre facultatif **-p** uniquement si vous souhaitez utiliser des options de configuration différentes de vos valeurs par défaut. Si vous ne spécifiez pas **-p**, les options de configuration définies dans le fichier `installation.properties` sont utilisées. Pour plus d'informations, voir «Options de configuration de MFT sur Multiplatforms», à la page 58.

-f

Facultatif. Force la commande à désenregistrer l'agent du gestionnaire de file d'attente de coordination même si les fichiers de configuration de l'agent sont introuvables. Etant donné que les informations sur le gestionnaire de files d'attente de l'agent ne sont pas disponibles dans cette situation, la commande se connecte directement au gestionnaire de files d'attente de coordination au lieu d'utiliser le gestionnaire de files d'attente de l'agent comme elle le ferait normalement.

-mquserid (*userID*)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de files d'attente de l'agent, sauf si le paramètre **-f** forcé est présent. Si le paramètre **-f** est présent, il indique l'ID utilisateur à utiliser pour l'authentification auprès du gestionnaire de files d'attente de coordination.

-mqpassword (*mot de passe*)

Facultatif. Indique le mot de passe à utiliser pour l'authentification auprès du gestionnaire de files d'attente de l'agent, sauf si le paramètre **-f** forcé est présent. Si le paramètre **-f** est présent, il indique le mot de passe à utiliser pour l'authentification auprès du gestionnaire de file d'attente

de coordination. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

Agent_Name

Obligatoire. Nom de l'agent à supprimer.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, AGENT3 et sa configuration sur le gestionnaire de files d'attente de coordination QM_COORD1 sont supprimés:

```
fteDeleteAgent -p QM_COORD1 AGENT3
```

Cet exemple de commande génère les commandes MQSC suivantes pour supprimer les trois files d'attente de l'agent:

```
CLEAR QLOCAL(SYSTEM.FTE.COMMAND.AGENT3)
DELETE QLOCAL(SYSTEM.FTE.COMMAND.AGENT3)
CLEAR QLOCAL(SYSTEM.FTE.DATA.AGENT3)
DELETE QLOCAL(SYSTEM.FTE.DATA.AGENT3)
CLEAR QLOCAL(SYSTEM.FTE.REPLY.AGENT3)
DELETE QLOCAL(SYSTEM.FTE.REPLY.AGENT3)
CLEAR QLOCAL(SYSTEM.FTE.STATE.AGENT3)
DELETE QLOCAL(SYSTEM.FTE.STATE.AGENT3)
CLEAR QLOCAL(SYSTEM.FTE.EVENT.AGENT3)
DELETE QLOCAL(SYSTEM.FTE.EVENT.AGENT3)
CLEAR QLOCAL(SYSTEM.FTE.AUTHADM1.AGENT3)
DELETE QLOCAL(SYSTEM.FTE.AUTHADM1.AGENT3)
CLEAR QLOCAL(SYSTEM.FTE.AUTHAGT1.AGENT3)
DELETE QLOCAL(SYSTEM.FTE.AUTHAGT1.AGENT3)
CLEAR QLOCAL(SYSTEM.FTE.AUTHTRN1.AGENT3)
DELETE QLOCAL(SYSTEM.FTE.AUTHTRN1.AGENT3)
CLEAR QLOCAL(SYSTEM.FTE.AUTHOPS1.AGENT3)
DELETE QLOCAL(SYSTEM.FTE.AUTHOPS1.AGENT3)
CLEAR QLOCAL(SYSTEM.FTE.AUTHSCH1.AGENT3)
DELETE QLOCAL(SYSTEM.FTE.AUTHSCH1.AGENT3)
CLEAR QLOCAL(SYSTEM.FTE.AUTHMON1.AGENT3)
DELETE QLOCAL(SYSTEM.FTE.AUTHMON1.AGENT3)
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Référence associée

[«fteStopAgent», à la page 544](#)

Utilisez la commande **fteStopAgent** pour arrêter un agent Managed File Transfer de manière contrôlée ou pour arrêter un agent immédiatement si nécessaire à l'aide du paramètre **-i**.

[«Agent fteClean: nettoyage d'un agent MFT», à la page 394](#)

La commande **fteCleanAgent** permet de nettoyer les files d'attente utilisées par un Managed File Transfer Agent en supprimant les messages des files d'attente persistantes et non persistantes utilisées par l'agent. Utilisez la commande **fteCleanAgent** si vous rencontrez des problèmes lors du démarrage d'un agent, ce qui peut être dû aux informations restant dans les files d'attente utilisées par l'agent.

[«Agent fteCreate\(créer un agent MFT\)», à la page 399](#)

La commande **fteCreateAgent** crée un Managed File Transfer Agent et sa configuration associée.

[«fteStartAgent», à la page 540](#)

La commande **fteStartAgent** démarre un agent Managed File Transfer à partir de la ligne de commande.

fteDeleteLogger

La commande **fteDeleteLogger** permet de supprimer un consigneur Managed File Transfer et sa configuration. Les fichiers journaux existants associés au consigneur peuvent être conservés ou supprimés.

Important :  Sous IBM MQ for UNIX, Linux, and Windows, seuls les utilisateurs qui sont des administrateurs d'IBM MQ (et membres du groupe mqm) peuvent exécuter cette commande. Si vous tentez d'exécuter cette commande en tant qu'utilisateur qui n'est pas un administrateur IBM MQ, le message d'erreur BFGCL0502E: Vous n'êtes pas autorisé à effectuer l'opération demandée. et la commande n'est pas exécutée.

 Sur les systèmes z/OS, l'utilisateur doit remplir (au moins) l'une des conditions suivantes pour pouvoir exécuter la commande migrate :

- Etre membre du groupe mqm (si le groupe mqm est défini sur le système).
-  Etre membre du groupe désigné dans la variable d'environnement BFG_GROUP_NAME (si un tel groupe a été désigné).
-  Ne pas avoir de valeur définie dans la variable d'environnement BFG_GROUP_NAME lorsque la commande est exécutée.

Objet

Arrêtez le consigneur à l'aide de la commande **fteStopLogger** avant d'exécuter la commande **fteDeleteLogger**.

Si vous avez configuré votre consigneur pour qu'il s'exécute en tant que service Windows, l'exécution de la commande **fteDeleteLogger** supprime la définition de service.

Le répertoire de configuration du consigneur contient un script MQSC permettant de supprimer les files d'attente et l'abonnement du consigneur. Ces files d'attente sont les suivantes:

- SYSTEME SYSTEM.FTE.LOG.CMD.*nom_journal*
- SYSTEME SYSTEM.FTE.LOG.RJCT.*nom_connexion*

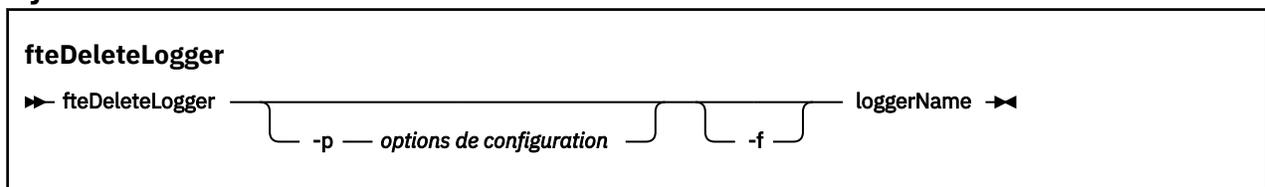
Le nom de l'abonnement est le suivant:

- SYSTEME SYSTEM.FTE.AUTO.SUB.*nom_connexion*

Le script MQSC est disponible à l'adresse

`MQ_DATA_PATH\mqft\config\coordination_qmgr\loggers\logger_name\logger_name_delete.mqsc`

Syntaxe



Paramètres

-p (*options_configuration*)

Facultatif. Détermine l'ensemble des options de configuration utilisées pour démarrer le consigneur de base de données autonome. Utilisez le nom d'un ensemble d'options de configuration comme

valeur pour le paramètre **-p** . Par convention, cette valeur correspond au nom d'un gestionnaire de files d'attente de coordination. Si vous ne spécifiez pas ce paramètre, l'ensemble d'options de configuration par défaut est utilisé.

-f

Facultatif. Force la suppression des fichiers journaux créés par ce consigneur. Si ce paramètre est omis, tous les fichiers journaux créés par le consigneur sont conservés et doivent être supprimés manuellement lorsqu'ils ne sont plus nécessaires.

nom_consignateur

Obligatoire. Nom du consigneur à supprimer.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, un consigneur appelé `logger1` est supprimé. Le paramètre **-f** a été spécifié, ce qui entraîne la suppression des fichiers journaux du consigneur ainsi que des fichiers de configuration du consigneur.

```
fteDeleteLogger -f logger1
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Référence associée

«[fteCreateLogger \(création d'un fichier MFT ou d'un consigneur de base de données\)](#)», à la page 418
Utilisez la commande **fteCreateLogger** pour créer un fichier Managed File Transfer ou un consigneur de base de données.

«[fteStartLogger](#)», à la page 542

La commande **fteStartLogger** démarre une application de journalisation Managed File Transfer .

«[fteStopLogger](#)», à la page 546

La commande **fteStopLogger** arrête un consigneur Managed File Transfer .

«[fteModifyLogger \(exécutez un consigneur MFT en tant que service Windows \)](#)», à la page 504

La commande **fteModifyLogger** permet de modifier un consigneur Managed File Transfer afin qu'il puisse être exécuté en tant que service Windows . Vous pouvez utiliser cette commande uniquement sur les plateformes Windows , vous devez être exécuté par un utilisateur qui est un administrateur IBM MQ et un membre du groupe mqm, et vous devez d'abord arrêter le consigneur à l'aide de la commande **fteStopLogger** .

Moniteur fteDelete: suppression d'un moniteur de ressources MFT

La commande **fteDeleteMonitor** permet d'arrêter et de supprimer un moniteur de ressources Managed File Transfer existant à l'aide de la ligne de commande. Exécutez cette commande sur l'agent de surveillance des ressources.

Objet

La commande **fteDeleteMonitor** permet d'arrêter la surveillance d'une ressource et de supprimer la définition du moniteur de l'agent de surveillance. Lorsque vous exécutez cette commande, aucune autre interrogation de la ressource n'est effectuée et aucune autre tâche n'est démarrée.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, le moniteur de ressources MONITOR1 avec une surveillance (et un agent source de transfert de fichier) AGENT1 est supprimé:

```
fteDeleteMonitor -ma AGENT1 -mm QM_JUPITER -mn MONITOR1
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Tâches associées

[«Surveillance des ressources MFT», à la page 174](#)

Vous pouvez surveiller les ressources Managed File Transfer ; par exemple, une file d'attente ou un répertoire. Lorsqu'une condition est satisfaite sur cette ressource, le moniteur de ressources démarre une tâche, telle qu'un transfert de fichier. Vous pouvez créer un moniteur de ressources à l'aide de la commande **fteCreateMonitor** ou de la vue **Moniteurs** dans le plug-in Managed File Transfer pour IBM MQ Explorer.

Référence associée

[«fteCreateMonitor: créer un moniteur de ressources MFT», à la page 424](#)

La commande **fteCreateMonitor** crée et démarre un nouveau moniteur de ressources à partir de la ligne de commande. Vous pouvez surveiller une ressource (par exemple, le contenu d'un répertoire) à l'aide de Managed File Transfer de sorte que lorsqu'une condition de déclenchement est remplie, une tâche spécifiée, telle qu'un transfert de fichier, est démarrée.

[«fteListMoniteurs: répertorie les moniteurs de ressources MFT», à la page 487](#)

La commande **fteListMonitors** permet de répertorier tous les moniteurs de ressources existants dans un réseau Managed File Transfer à l'aide de la ligne de commande.

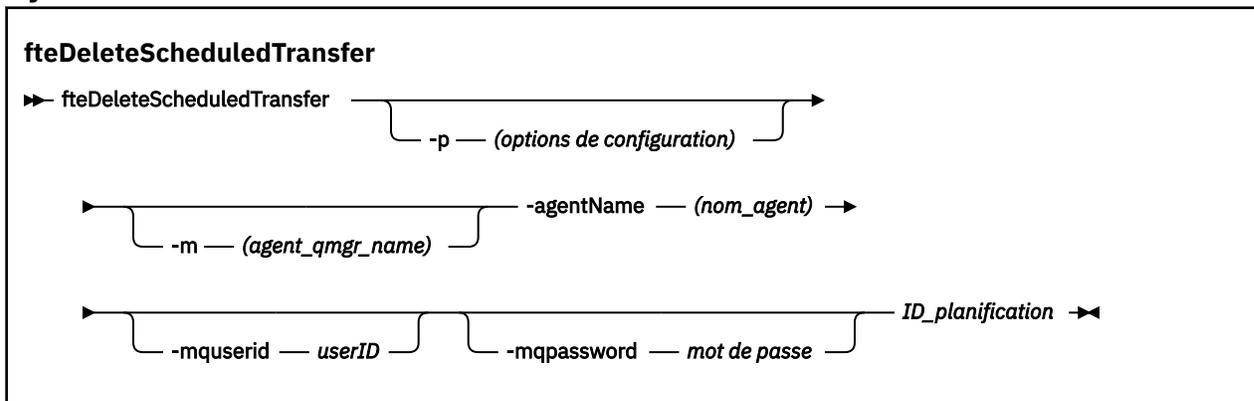
fteDeleteScheduledTransfer

Objet

La commande **fteDeleteScheduledTransfer** permet de supprimer un transfert planifié Managed File Transfer que vous avez précédemment créé à l'aide de la ligne de commande ou du IBM MQ Explorer.

Indiquez le paramètre facultatif **-p** pour cette commande uniquement si vous souhaitez utiliser des options de configuration différentes de celles par défaut. Si vous ne spécifiez pas **-p**, les options de configuration définies dans `installation.properties` sont utilisées. Pour plus d'informations, voir [«Options de configuration de MFT sur Multiplatforms», à la page 58](#).

Syntaxe



Paramètres

-p (*options_configuration*)

Facultatif. Si vous disposez de plusieurs gestionnaires de files d'attente de coordination, utilisez ce paramètre pour spécifier explicitement le transfert planifié à supprimer. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite les options de configuration associées à ce gestionnaire de files d'attente de coordination non défini par défaut.

Si vous ne spécifiez pas ce paramètre, les options de configuration basées sur le gestionnaire de files d'attente de coordination par défaut sont utilisées.

-m (*agent_qmgr_name*)

Facultatif. Nom du gestionnaire de files d'attente auquel l'agent source est connecté. Si vous ne spécifiez pas ce paramètre, le gestionnaire de files d'attente de l'agent est déterminé à partir des options de configuration utilisées.

-agentName (*nom_agent*)

Obligatoire. Nom de l'agent source duquel vous souhaitez supprimer le transfert planifié.

-mquserid (*userID*)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de files d'attente de commandes.

-mqpassword (*mot de passe*)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de commandes. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

ID_planification

Obligatoire. ID du transfert planifié à supprimer.

Vous pouvez trouver l'ID de planification en exécutant la commande [fteListScheduledTransfers](#) sur le nom de l'agent source.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, un transfert planifié sur l'agent source AGENT2 avec l'ID 27 est supprimé:

```
fteDeleteScheduledTransfer -agentName AGENT2 27
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Tâches associées

«Création d'un transfert de fichiers planifié», à la page 166

Vous pouvez planifier un nouveau transfert de fichiers à partir de l'explorateur IBM MQ Explorer ou de la ligne de commande. Le transfert planifié peut contenir un seul fichier ou plusieurs fichiers dans un groupe. Vous pouvez effectuer un transfert de fichier planifié une seule fois ou répéter le transfert plusieurs fois.

Référence associée

«fteListScheduledTransfers», à la page 490

La commande **fteListScheduledTransfers** permet de répertorier tous les transferts Managed File Transfer que vous avez précédemment créés à l'aide de la ligne de commande ou du IBM MQ Explorer.

fteDeleteTemplates

La commande **fteDeleteTemplates** permet de supprimer un modèle Managed File Transfer existant d'un gestionnaire de files d'attente de coordination.

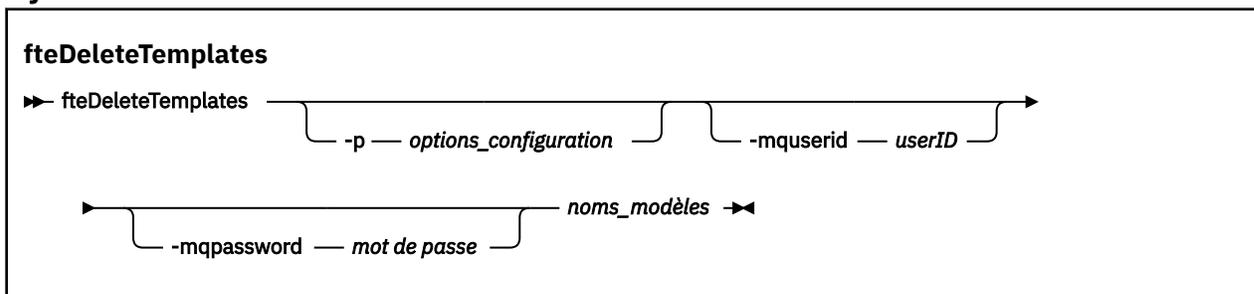
Objet

La commande **fteDeleteTemplates** supprime un ou plusieurs modèles de transfert de fichier d'un gestionnaire de files d'attente de coordination. Lorsque vous exécutez cette commande, une demande est transmise au système IBM MQ pour supprimer les modèles du gestionnaire de files d'attente de coordination afin que les modèles ne soient plus disponibles pour l'explorateur IBM MQ ou la ligne de commande. Les modèles que vous supprimez peuvent continuer à être accessibles pendant un bref intervalle après la fin de la commande jusqu'à ce que le système IBM MQ exécute la demande.

Vous pouvez exécuter la commande **fteDeleteTemplates** à partir de n'importe quel système pouvant se connecter au réseau IBM MQ, puis l'acheminer vers le gestionnaire de files d'attente de coordination. Pour que la commande puisse être exécutée, vous devez avoir installé Managed File Transfer sur ce système et avoir configuré le Managed File Transfer de ce système pour communiquer avec le réseau IBM MQ. Si aucun détail de connectivité n'est disponible, les détails du gestionnaire de files d'attente de l'agent sont utilisés à la place pour la connexion, à condition que ces détails soient disponibles.

Spécifiez le paramètre facultatif **-p** pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de votre ensemble par défaut. Pour plus d'informations, voir [Options de configuration](#).

Syntaxe



Paramètres

-p (options_configuration)

Facultatif. Ce paramètre détermine l'ensemble des options de configuration à utiliser pour supprimer le modèle. Par convention, utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

-mquserid (userID)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de file d'attente de coordination.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de coordination. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

(noms_modèle)

Obligatoire. Indiquez un ou plusieurs noms de modèle à supprimer. Indiquez le nom tel qu'il est affiché par la commande **fteListTemplates**.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, le modèle STANDBY est supprimé:

```
fteDeleteTemplates STANDBY
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Tâches associées

«Utilisation des modèles de transfert de fichiers», à la page 200

Vous pouvez utiliser des modèles de transfert de fichiers pour stocker des paramètres de transfert de fichiers communs pour des transferts répétés ou complexes. Créez un modèle de transfert à partir de la ligne de commande à l'aide de la commande **fteCreateTemplate** ou utilisez la IBM MQ Explorer pour créer un modèle de transfert à l'aide de l'assistant **Créer un modèle pour Managed File Transfer**, ou sauvegardez un modèle lors de la création d'un transfert de fichier en cochant la case **Sauvegarder les paramètres de transfert en tant que modèle**. La fenêtre **Modèles de transfert** affiche tous les modèles de transfert que vous avez créés dans votre réseau Managed File Transfer.

«Création d'un modèle de transfert de fichiers à l'aide de IBM MQ Explorer», à la page 202

Vous pouvez créer un modèle de transfert de fichier à partir de IBM MQ Explorer ou de la ligne de commande. Vous pouvez ensuite utiliser ce modèle pour créer de nouveaux transferts de fichiers à l'aide des détails du modèle ou soumettre le modèle pour démarrer le transfert de fichiers.

Référence associée

«fteCreateTemplate: créer un nouveau modèle de transfert de fichiers», à la page 431

La commande **fteCreateTemplate** crée un modèle de transfert de fichier que vous pouvez conserver pour une utilisation ultérieure. Le seul paramètre obligatoire est le paramètre **-tn (template_name)**. Tous

les autres paramètres sont facultatifs, bien que si vous spécifiez une spécification de fichier source, vous devez également fournir un fichier de destination. De même, si vous spécifiez un fichier de destination, vous devez également spécifier une spécification de fichier source.

«[fteListTemplates](#)», à la page 492

La commande **fteListTemplates** permet de répertorier les modèles de transfert Managed File Transfer disponibles sur un gestionnaire de files d'attente de coordination.

fteDisplayVersion

Utilisez la commande **fteDisplayVersion** pour afficher la version de Managed File Transfer que vous avez installée.

Objet

Un technicien de maintenance IBM peut vous demander d'exécuter la commande **fteDisplayVersion** pour vous aider à identifier les problèmes.

Syntaxe



Paramètres

-v

Facultatif. Affiche une quantité prolixe d'informations sur la version du produit.

Les détails précis affichés lorsque vous spécifiez le paramètre **-v** peuvent varier d'une édition de produit à l'autre. Il n'est pas recommandé de s'appuyer sur des informations spécifiques disponibles dans la sortie de la commande `fteDisplayVersion -v`.

 Sous z/OS, **-v** affiche la valeur de la propriété **productId**, si l'ID produit a été spécifié.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple sans paramètres spécifiés

Dans cet exemple, la commande **fteDisplayVersion** est spécifiée sans aucun paramètre.

```
fteDisplayVersion
```

La sortie de cette commande correspond au niveau de version du produit. Par exemple, voici la sortie pour IBM MQ 9.0:

```
5655-MFT, 5724-H72 Copyright IBM Corp. 2008, 2016. ALL RIGHTS RESERVED
IBM MQ Components:
Name:          IBM MQ Managed File Transfer
Version:       9.0.0.0
```

Exemple avec le paramètre -v spécifié

Dans cet exemple, la commande **fteDisplayVersion** est spécifiée avec le paramètre **-v**.

```
fteDisplayVersion -v
```

La sortie de cette commande inclut des informations plus détaillées sur la version du produit. Exemple :

```
5655-MFT, 5724-H72 Copyright IBM Corp. 2008, 2016. ALL RIGHTS RESERVED
IBM MQ Components:
Name:          IBM MQ Managed File Transfer
Version:       9.0.0.0
Level:         p900-L160512.4
Platform:     Windows 10 (10.0)
Architecture: amd64
JVM:          JRE 1.8.0 Windows 10 amd64-64 Compressed References 20160427_301573 (JIT enabled, AOT
enabled)
              J9VM - R28_Java8_SR3_20160427_1620_B301573
              JIT  - tr.r14.java.green_20160329_114288
              GC   - R28_Java8_SR3_20160427_1620_B301573_CMPRSS
              J9CL - 20160427_301573
Product:       C:\Program Files\IBM\MQ_1
Configuration: C:\ProgramData\IBM\MQ\mqft

Name:          IBM MQ JMS Provider
Version:       9.0.0.0
Level:         p900-L160512.4

Name:          IBM MQLight Service for Bluemix JMS Provider
Version:
Level:         p900-L160512.4

Name:          Common Services for Java Platform, Standard Edition
Version:       9.0.0.0
Level:         p900-L160512.4

Name:          Java Message Service Client
Version:       9.0.0.0
Level:         p900-L160512.4

Name:          IBM MQ classes for Java Message Service
Version:       9.0.0.0
Level:         p900-L160512.4

Name:          IBM MQ classes for Java
Version:       9.0.0.0
Level:         p900-L160512.4
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Agents **fteList**: répertorie les agents MFT pour un gestionnaire de files d'attente de coordination

La commande **fteListAgents** permet de répertorier tous les agents Managed File Transfer enregistrés auprès d'un gestionnaire de files d'attente de coordination particulier.

Objet

Vous pouvez exécuter la commande **fteListAgents** à partir de n'importe quel système pouvant se connecter au gestionnaire de file d'attente de coordination. Les détails suivants de chaque agent sont dirigés vers l'unité de sortie standard (STDOUT):

- Nom de l'agent

- Gestionnaire de files d'attente d'agent
- Si l'agent est un agent de pont de protocole, le nom de l'agent est ajouté à `bridge` .
- Si l'agent est un agent de pont Connect:Direct , le nom de l'agent est ajouté à (`Connect:Direct` `bridge`)
- Statut de l'agent

Cette commande utilise le fichier `coordination.properties` pour se connecter au gestionnaire de files d'attente de coordination. Pour plus d'informations, voir la section «[Le fichier MFT coordination.properties](#)», à la page 553.

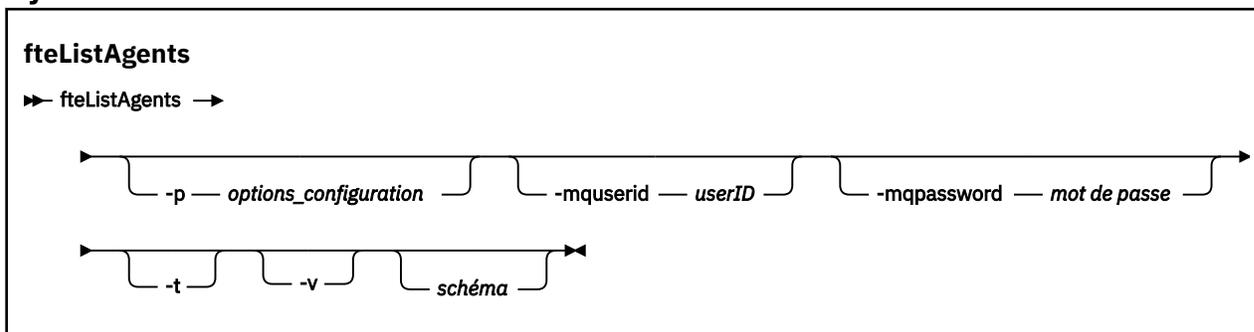
Spécifiez le paramètre facultatif **-p** pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de votre ensemble par défaut. Pour plus d'informations, voir la section «[Options de configuration de MFT sur Multiplatforms](#)», à la page 58.

Si un agent n'est pas répertorié par la commande **fteListAgents** , utilisez le diagramme de diagnostic de la rubrique suivante pour localiser et résoudre le problème: [Si votre agent n'est pas répertorié par la commande fteListAgents](#).

Informations sur le statut de l'agent

Les informations de statut d'agent générées par cette commande sont générées à partir des messages de statut publiés par l'agent dans `SYSTEM.FTE` . Ces messages sont décrits dans la rubrique «[Format de message d'état de l'agent MFT](#)», à la page 757. Les informations de statut générées par la commande **fteListAgents** indiquent le statut de l'agent au moment de la publication du dernier message de statut. La fréquence de ces messages d'état dépend de la valeur de la propriété `agentStatusPublishRateLimit`. Pour plus de détails sur cette propriété, voir la rubrique «[Le fichier MFT agent.properties](#)», à la page 564.

Syntaxe



Paramètres

-p (options_configuration)

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour émettre la demande de listage des agents. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

-mquserid (userID)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de file d'attente de coordination.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de coordination. Vous devez également spécifier le paramètre **-mquserid** . Si vous spécifiez

-mquserid, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

-v

Facultatif. Spécifie le mode prolix. Le mode prolix génère une sortie supplémentaire pour chaque agent, y compris le nombre actuel de transferts sous la forme Source/Destination, où Source est le nombre actuel de transferts source et Destination est le nombre actuel de transferts cible.

Les informations de transfert en cours sont obtenues à partir de la publication de statut de l'agent, qui est décrite dans la rubrique suivante: [«Format de message d'état de l'agent MFT»](#), à la page 757. Par conséquent, ces informations de transfert ne sont précises que dans le cadre du paramètre de la valeur de la propriété [agentStatusPublishRateLimit](#) de l'agent (qui, par défaut, est de 30 secondes).

-t

Facultatif. Spécifie le mode concis. Dans IBM MQ 9.0.2, la sortie inclut la colonne **Status Age** par défaut. Si vous ne souhaitez pas voir les informations **Status Age**, vous pouvez exécuter la commande avec le paramètre **-t** pour masquer la colonne. Pour plus d'informations, voir [Que faire si la commande **fteListAgents** affiche le statut d'agent UNKNOWN.](#)

schéma

Facultatif. Modèle à utiliser pour filtrer la liste des agents Managed File Transfer. Ce modèle est mis en correspondance avec le nom de l'agent. Les caractères astérisque (*) sont interprétés comme des caractères génériques, qui correspondent à n'importe quelle valeur, y compris les caractères zéro.

Sur les systèmes UNIX et Linux, vous devez mettre en échappement les caractères spéciaux tels que l'astérisque (*) et le signe dièse (#) avec des guillemets (") ou des guillemets (") si vous voulez qu'ils soient traités comme des littéraux. Si vous ne mettez pas ces caractères en échappement, ils sont interprétés en fonction de leur signification sur le système UNIX ou Linux spécifique.

Si vous ne spécifiez pas ce paramètre, tous les agents enregistrés auprès du gestionnaire de file d'attente de coordination sont répertoriés.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, tous les agents enregistrés sur le gestionnaire de files d'attente et détaillés dans les options de configuration dont le nom commence par B sont répertoriés:

```
fteListAgents "B*"
```

Dans cet exemple, les agents enregistrés auprès du gestionnaire de files d'attente de coordination QM_EUROPE (gestionnaire de files d'attente de coordination autre que par défaut) sont répertoriés en mode prolix:

```
fteListAgents -p QM_EUROPE -v
```

La sortie de cette commande est la suivante:

Agent Name:	Queue Manager Name:	Transfers: (Source/Destination)	Status:
BERLIN	QM_BERLIN	7/0	RUNNING
LONDON	QM_LONDON	0/0	RUNNING
MADRID	QM_MADRID	0/1	UNREACHABLE

Pour obtenir la liste des valeurs de statut d'agent possibles et leur signification, voir la rubrique [«Valeurs de statut de l'agent MFT»](#), à la page 642.

Dans cet exemple, tous les agents enregistrés auprès du gestionnaire de file d'attente de coordination et dont le nom commence par BRIDGE sont répertoriés en mode prolix:

```
fteListAgents -v "BRIDGE*"
```

La sortie de cette commande est la suivante:

```
C:\Program Files\IBM\WMQFTE\bin>fteListAgents -v
5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
Agent Name:                               Queue Manager Name:   Transfers:   Status:
                                           (Source/Destination)
BRIDGE_FTP ( bridge )                     QM_JUPITER            0/0         STOPPED
BRIDGE_CD1 (Connect:Direct bridge)        QM_JUPITER            0/0         STOPPED
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Tâches associées

«Liste des agents MFT», à la page 217

Vous pouvez répertorier les agents Managed File Transfer enregistrés auprès d'un gestionnaire de files d'attente particulier à l'aide de la ligne de commande ou de l' IBM MQ Explorer.

Référence associée

«Valeurs de statut de l'agent MFT», à la page 642

Les commandes **fteListAgents** et **fteShowAgentDetails** produisent des informations sur le statut de l'agent. Il existe plusieurs valeurs possibles pour ce statut.

«fteShowAgentDetails», à la page 530

La commande **fteShowAgentDetails** permet d'afficher les détails d'un Managed File Transfer Agent particulier. Il s'agit des détails stockés par le gestionnaire de files d'attente de coordination Managed File Transfer de l'agent.

fteListMoniteurs: répertorie les moniteurs de ressources MFT

La commande **fteListMonitors** permet de répertorier tous les moniteurs de ressources existants dans un réseau Managed File Transfer à l'aide de la ligne de commande.

Objet

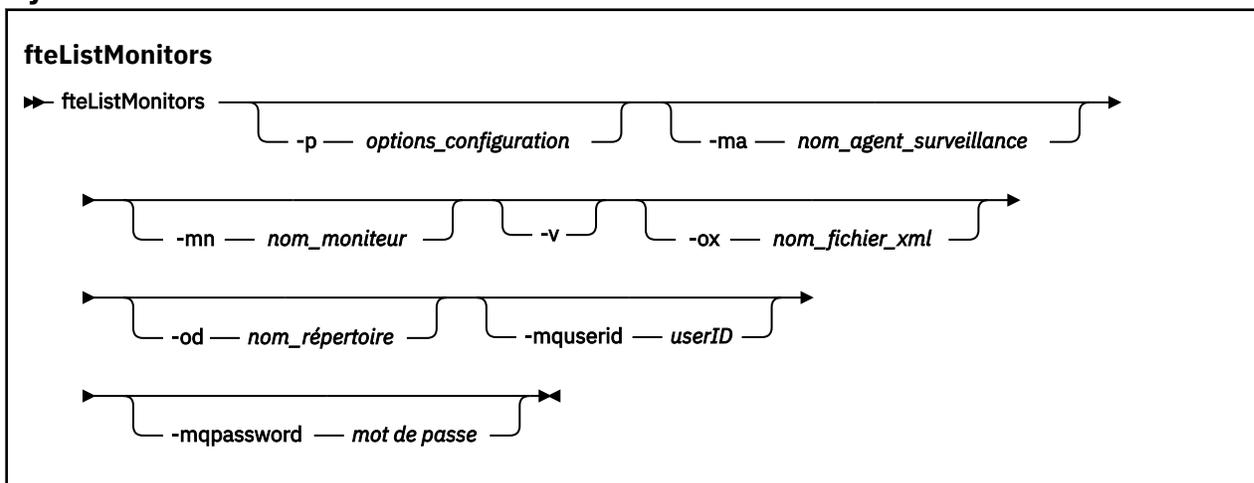
La commande **fteListMonitors** répertorie les moniteurs de ressources existants. Vous pouvez filtrer le résultat de la commande en spécifiant un nom d'agent et un nom de moniteur de ressources.

Cette commande utilise le fichier `coordination.properties` pour se connecter au gestionnaire de files d'attente de coordination. Pour plus d'informations, voir la section «[Le fichier MFT coordination.properties](#)», à la page 553.

Vous pouvez utiliser le paramètre **-ox** pour exporter un moniteur de ressources vers un fichier XML. Pour plus d'informations sur l'utilisation de ce fichier XML, voir «[fteCreateMonitor: créer un moniteur de ressources MFT](#)», à la page 424.

Spécifiez le paramètre facultatif **-p** pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de votre ensemble par défaut. Pour plus d'informations, voir [Options de configuration](#).

Syntaxe



Paramètres

-p (options_configuration)

Facultatif. Ce paramètre détermine l'ensemble des options de configuration à utiliser pour annuler le transfert. Par convention, utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination non défini par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

-ma (nom_agent_surveillance)

Facultatif. Filtre les moniteurs de ressources par nom d'agent en utilisant le modèle que vous fournissez en entrée. Les caractères astérisque (*) sont interprétés comme des caractères génériques qui correspondent à zéro ou plusieurs caractères. Si vous ne spécifiez pas le paramètre **-ma**, tous les moniteurs de ressources associés à tous les agents du gestionnaire de files d'attente de coordination par défaut sont répertoriés par défaut.

-mn (nom_moniteur)

Facultatif. Filtre les moniteurs de ressources par nom de moniteur à l'aide du modèle que vous fournissez en entrée. Les caractères astérisque (*) sont interprétés comme des caractères génériques qui correspondent à zéro ou plusieurs caractères. Si vous ne spécifiez pas le paramètre **-mn**, tous les moniteurs de ressources associés à tous les agents du gestionnaire de files d'attente de coordination par défaut sont répertoriés par défaut.

-mquserid (userID)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de file d'attente de coordination.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de coordination. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous êtes invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

-v

Facultatif. Génère une sortie prolixe qui inclut des informations supplémentaires sur l'état du moniteur, notamment si le moniteur est démarré ou arrêté, le chemin d'accès à la ressource de répertoire surveillée et les conditions de déclenchement.

-ox (nom_fichier_xml)

Facultatif. Vous devez spécifier ce paramètre en combinaison avec les paramètres **-ma** et **-mn**. Exporte le moniteur de ressources dans un fichier XML qui peut ensuite être utilisé par la commande **fteCreateMonitor** et le paramètre **-ix**.

V 9.0.5

Le paramètre **-ox** ne doit pas être combiné avec le paramètre **-od**.

V 9.0.5

-od (nom_répertoire)

Facultatif. Exporte plusieurs définitions de moniteur de ressources dans le répertoire indiqué. Chaque définition de moniteur de ressources est sauvegardée dans un fichier XML distinct avec un nom au format *agent name.monitor name.xml*. Vous devez spécifier un répertoire cible valide pour les fichiers XML, sinon un message d'erreur s'affiche. Ce paramètre ne doit pas être combiné avec le paramètre **-ox**.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple: affichage de la liste des moniteurs de ressources

Dans cet exemple, tous les moniteurs de ressources associés à l'agent de surveillance (et à l'agent source pour les transferts de fichiers associés au moniteur) AGENT1 sont répertoriés:

```
fteListMonitors -ma AGENT1
```

Exemple: exportation d'un moniteur de ressources dans un fichier XML

Dans cet exemple, un seul moniteur de ressources, MONITOR1, sur AGENT1, est exporté vers le fichier XML *filename1.xml* en spécifiant un nom de fichier XML avec le paramètre **-ox**:

```
fteListMonitors -ma AGENT1 -mn MONITOR1 -ox filename1.xml
```

Exemple: exportation d'un moniteur de ressources vers un répertoire spécifié

V 9.0.5

Dans cet exemple, un seul moniteur de ressources, MONITOR1, sur AGENT1 est exporté vers le répertoire spécifié par le paramètre **-od**. A l'exception de la différence dans le format de nom de fichier XML, cet exemple est similaire à l'utilisation du paramètre **-ox**.

```
fteListMonitors -ma AGENT1 -mn MONITOR1 -od /usr/mft/resmonbackup
```

Exemples: exportation d'un lot de moniteurs de ressources dans un fichier XML dans un répertoire spécifié

V 9.0.5

Dans tous les exemples suivants, les moniteurs de ressources sont exportés vers le répertoire spécifié par le paramètre **-od**. Chaque définition de moniteur de ressources est sauvegardée dans un fichier XML distinct avec un nom au format *agent name.monitor name.xml*.

Dans cet exemple, tous les moniteurs de ressources sont exportés vers le répertoire spécifié:

```
fteListMonitors -od /usr/mft/resmonbackup
```

Dans cet exemple, tous les moniteurs de ressources sur AGENT1 sont exportés vers le répertoire spécifié:

```
fteListMonitors -ma AGENT1 -od /usr/mft/resmonbackup
```

Vous pouvez utiliser la correspondance des caractères génériques pour définir les moniteurs de ressources à exporter à l'aide d'un astérisque (*) lorsque vous spécifiez un modèle à mettre en correspondance avec des noms d'agent ou des noms de moniteur, ou les deux.

Dans cet exemple, tous les moniteurs de ressources sur AGENT1 dont les noms correspondent au modèle MON* sont exportés vers le répertoire spécifié:

```
fteListMonitors -ma AGENT1 -mn MON* -od /usr/mft/resmonbackup
```

Dans cet exemple, tous les moniteurs de ressources sur les agents dont les noms correspondent au modèle AGEN* sont exportés dans le répertoire spécifié:

```
fteListMonitors -ma AGEN* -od /usr/mft/resmonbackup
```

Dans cet exemple, tous les moniteurs de ressources dont les noms correspondent au modèle MON* sur les agents dont les noms correspondent au modèle AGENT* sont exportés vers le répertoire spécifié:

```
fteListMonitors -ma AGENT* -mn MON* -od /usr/mft/resmonbackup
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Tâches associées

«Surveillance des ressources MFT», à la page 174

Vous pouvez surveiller les ressources Managed File Transfer ; par exemple, une file d'attente ou un répertoire. Lorsqu'une condition est satisfaite sur cette ressource, le moniteur de ressources démarre une tâche, telle qu'un transfert de fichier. Vous pouvez créer un moniteur de ressources à l'aide de la commande **fteCreateMonitor** ou de la vue **Moniteurs** dans le plug-in Managed File Transfer pour IBM MQ Explorer.

«Sauvegarde et restauration des moniteurs de ressources MFT», à la page 199

Vous pouvez sauvegarder les moniteurs de ressources que vous souhaitez pouvoir utiliser ultérieurement en exportant leurs définitions dans un fichier XML que vous pouvez ensuite importer pour créer un nouveau moniteur de ressources à partir de la sauvegarde.

Référence associée

«fteCreateMonitor: créer un moniteur de ressources MFT», à la page 424

La commande **fteCreateMonitor** crée et démarre un nouveau moniteur de ressources à partir de la ligne de commande. Vous pouvez surveiller une ressource (par exemple, le contenu d'un répertoire) à l'aide de Managed File Transfer de sorte que lorsqu'une condition de déclenchement est remplie, une tâche spécifiée, telle qu'un transfert de fichier, est démarrée.

«Moniteur fteDelete: suppression d'un moniteur de ressources MFT», à la page 477

La commande **fteDeleteMonitor** permet d'arrêter et de supprimer un moniteur de ressources Managed File Transfer existant à l'aide de la ligne de commande. Exécutez cette commande sur l'agent de surveillance des ressources.

fteListScheduledTransfers

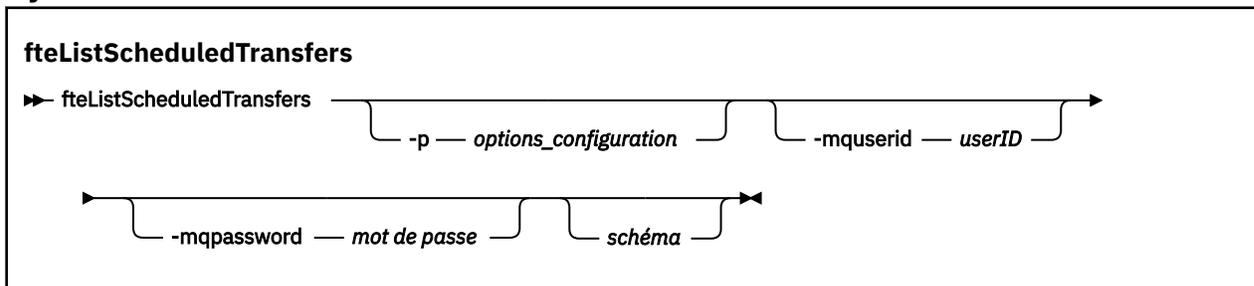
La commande **fteListScheduledTransfers** permet de répertorier tous les transferts Managed File Transfer que vous avez précédemment créés à l'aide de la ligne de commande ou du IBM MQ Explorer.

Objet

Vous pouvez répertorier tous les transferts planifiés en fonction des noms d'agent source ou du gestionnaire de files d'attente de coordination.

Indiquez le paramètre facultatif **-p** pour cette commande uniquement si vous souhaitez utiliser des options de configuration différentes de celles par défaut. Si vous ne spécifiez pas **-p**, les options de configuration définies dans `installation.properties` sont utilisées. Pour plus d'informations, voir «Options de configuration de MFT sur Multiplatforms», à la page 58.

Syntaxe



Paramètres

-p (options_configuration)

Facultatif. Si vous disposez de plusieurs gestionnaires de files d'attente de coordination, utilisez ce paramètre pour spécifier explicitement les agents pour lesquels vous souhaitez répertorier les transferts planifiés. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite les options de configuration associées à ce gestionnaire de files d'attente de coordination non défini par défaut.

Si vous ne spécifiez pas ce paramètre, les options de configuration basées sur le gestionnaire de files d'attente de coordination par défaut sont utilisées.

-mquserid (userID)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de file d'attente de coordination.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de coordination. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

schéma

Facultatif. Modèle à utiliser pour filtrer la liste des transferts planifiés Managed File Transfer. Ce modèle est mis en correspondance avec le nom de l'agent source. Les caractères astérisque (*) sont interprétés comme des caractères génériques qui correspondent à zéro ou plusieurs caractères.

Si vous ne spécifiez pas ce paramètre, tous les transferts planifiés enregistrés auprès du gestionnaire de file d'attente de coordination sont répertoriés par défaut.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, tous les transferts planifiés avec des agents source correspondant au modèle * 2 sont répertoriés:

```
fteListScheduledTransfers "*2"
```

Cet exemple de commande génère la sortie suivante. L'heure de début de la planification et l'heure du prochain transfert sont affichées en temps universel coordonné (UTC):

```
Schedule Identifier:      1
Source Agent Name:       AGENT2
Source File Name:        C:/export/Test/workspace/A.exe
Conversion Type:         binary
Destination File Name:   C:/import/Test/workspace/B001.zzx
Destination Agent Name:  AGENT1
Schedule Start Time:     2008-10-23T16:08+0100
Next Transfer:           2008-10-23T16:08+0100
Schedule Time Base:      source
```

Repeat Interval: minutes
Repeat Frequency: 1
Repeat Count: 30

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Tâches associées

«Création d'un transfert de fichiers planifié», à la page 166

Vous pouvez planifier un nouveau transfert de fichiers à partir de l'explorateur IBM MQ Explorer ou de la ligne de commande. Le transfert planifié peut contenir un seul fichier ou plusieurs fichiers dans un groupe. Vous pouvez effectuer un transfert de fichier planifié une seule fois ou répéter le transfert plusieurs fois.

Référence associée

«fteDeleteScheduledTransfer», à la page 479

fteListTemplates

La commande **fteListTemplates** permet de répertorier les modèles de transfert Managed File Transfer disponibles sur un gestionnaire de files d'attente de coordination.

Objet

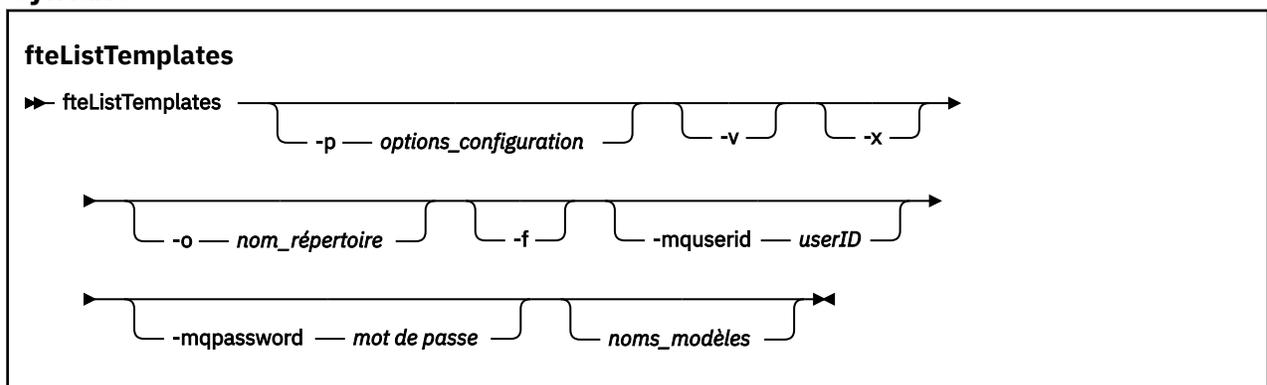
Cette commande répertorie tous les noms de modèle ou une sélection filtrée de noms de modèle. Le format de sortie de la liste peut être l'un des suivants:

- Noms de modèle uniquement (comportement par défaut)
- Noms de modèle avec un récapitulatif des modèles (mode prolix)
- Message XML complet décrivant les modèles (paramètres **-x** et **-o**)

Cette commande utilise le fichier `coordination.properties` pour se connecter au gestionnaire de files d'attente de coordination. Pour plus d'informations, voir la section «Le fichier MFT `coordination.properties`», à la page 553.

Spécifiez le paramètre facultatif **-p** pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de votre ensemble par défaut. Pour plus d'informations, voir [Options de configuration](#).

Syntaxe



Paramètres

-p

Facultatif. Ce paramètre détermine l'ensemble des options de configuration à utiliser pour supprimer le modèle. Par convention, utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

-v

Facultatif. Indique le mode prolix et fournit un bref récapitulatif de chaque modèle correspondant. Ce paramètre est ignoré si vous avez également spécifié le paramètre **-x**.

Le paramètre **-v** inclut un récapitulatif de chaque modèle. Exemple :

```
Template Name: STANDBY
Source Agent Name: AGENT1
Source QMgr: QM_JUPITER
Destination Agent Name: AGENT2
Destination QMgr: QM_NEPTUNE
Transfer Priority: 0
Transfer file specification
File Item Details
Mode: binary
Checksum: MD5
Source File:
  C:\payroll_reports\*.xls
Recursive: false
Disposition: leave
Destination File:
  C:\payroll_backup\*.xls
Type: file
Exist: error
```

Si vous ne spécifiez pas le paramètre **-v**, le mode de sortie par défaut consiste à répertorier les noms de modèles correspondants.

-x

Facultatif. Fournit un message au format XML pour chaque modèle correspondant. Ce paramètre est ignoré sauf si vous spécifiez également le paramètre **-o**.

-o (*nom_répertoire*)

Facultatif. Envoie le message XML formaté aux fichiers du répertoire nommé. Un fichier est créé pour chaque modèle et chaque fichier porte le même nom que le modèle avec un suffixe `.xml`. Ce paramètre est ignoré sauf si vous spécifiez également le paramètre **-x**.

-f

Facultatif. Force le remplacement de tout fichier de sortie existant. Ce paramètre est ignoré sauf si vous spécifiez également le paramètre **-o**. Si vous ne spécifiez pas **-f** mais que vous indiquez le nom d'un fichier de sortie existant, le comportement par défaut consiste à signaler une erreur et à continuer.

-mquserid (*userID*)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de file d'attente de coordination.

-mqpassword (*mot de passe*)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de coordination. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

(noms_modèle)

Facultatif. Liste d'un ou de plusieurs noms de modèle à répertorier. Un nom de modèle peut inclure un astérisque comme caractère générique correspondant à aucun ou plusieurs caractères. En fonction de votre système d'exploitation, vous devrez peut-être placer entre guillemets ("") les noms de modèle qui contiennent des caractères génériques ou des guillemets simples (") pour éviter l'extension de l'interpréteur de commandes. L'extension de l'interpréteur de commandes peut entraîner un comportement inattendu.

Si vous ne spécifiez rien pour *template_names*, la valeur par défaut est de répertorier tous les modèles.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, tous les modèles dont les noms commencent par ST sont répertoriés:

```
fteListTemplates "ST*"
```

Cet exemple crée le modèle STANDBY sous la forme d'un message au format XML dans le fichier STANDBY.xml du répertoire en cours:

```
fteListTemplates -x -o . STANDBY
```

Cette commande crée la sortie suivante dans STANDBY.xml:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <transferTemplate id="1864c1dd-ba02-4b34-bda9-dc6862448418" version="3.00">
  <name>STANDBY</name>
  <sourceAgentName>AGENT1</sourceAgentName>
  <sourceAgentQMgr>QM_JUPITER</sourceAgentQMgr>
  <sourceAgentQMgrHost>null</sourceAgentQMgrHost>
  <sourceAgentQMgrPort>-1</sourceAgentQMgrPort>
  <sourceAgentQMgrChannel>null</sourceAgentQMgrChannel>
  <destinationAgentName>AGENT2</destinationAgentName>
  <destinationAgentQMgr>QM_NEPTUNE</destinationAgentQMgr>
- <fileSpecs>
  - <item checksumMethod="MD5" mode="binary">
    - <source disposition="leave" recursive="false">
      <file>C:\payroll_reports\*.xls</file>
    </source>
    - <destination exist="error" type="file">
      <file>C:\payroll_backup\*.xls</file>
    </destination>
  </item>
</fileSpecs>
<priority>0</priority>
</transferTemplate>
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Agent fteMigrate: migration d'un agent FTE 7.0 vers MQ 7.5 ou version ultérieure

Si vous souhaitez migrer un agent existant et sa configuration associée depuis IBM WebSphere MQ File Transfer Edition 7.0 vers IBM WebSphere MQ 7.5 ou une version ultérieure, utilisez la commande **fteMigrateAgent** pour effectuer la migration. Cette commande peut être utilisée pour migrer un agent

standard, un agent Connect:Direct ou un agent de pont de protocole. La commande peut également être utilisée pour migrer plusieurs agents en une seule demande.

V 9.0.0 Depuis IBM MQ 9.0, Managed File Transfer ne prend pas en charge les agents Web. Si vous tentez d'utiliser la commande **fteMigrateAgent** pour migrer un agent Web à partir d'une version antérieure à IBM MQ 9.0, un message d'erreur s'affiche et indique que la migration d'un agent Web n'est pas pris en charge.

Remarque : Si vous migrez depuis IBM WebSphere MQ File Transfer Edition 7.0 ou une version ultérieure et que vous souhaitez continuer à utiliser la variable d'environnement FTE_CONFIG, vous pouvez le faire sans modifier la valeur de FTE_CONFIG. Vous pouvez effectuer une migration standard, mais BFG_DATA ne doit pas être défini et FTE_CONFIG doit être défini comme utilisé dans IBM WebSphere MQ 7.0.

Important : **ULW** Sous IBM MQ for UNIX, Linux, and Windows, seuls les utilisateurs qui sont des administrateurs d'IBM MQ (et membres du groupe mqm) peuvent exécuter cette commande. Si vous tentez d'exécuter cette commande en tant qu'utilisateur qui n'est pas un administrateur IBM MQ, le message d'erreur BFGCL0502E: Vous n'êtes pas autorisé à effectuer l'opération demandée. et la commande n'est pas exécutée.

z/OS Sur les systèmes z/OS, l'utilisateur doit remplir (au moins) l'une des conditions suivantes pour pouvoir exécuter la commande migrate :

- Etre membre du groupe mqm (si le groupe mqm est défini sur le système).
- **V 9.0.0.1** Etre membre du groupe désigné dans la variable d'environnement BFG_GROUP_NAME (si un tel groupe a été désigné).
- **V 9.0.0.1** Ne pas avoir de valeur définie dans la variable d'environnement BFG_GROUP_NAME lorsque la commande est exécutée.

Si votre agent est configuré pour s'exécuter en tant que service Windows, utilisez la commande **fteModifyAgent** pour reconfigurer l'agent afin qu'il ne soit plus un service Windows. Une fois la migration terminée, utilisez à nouveau la commande **fteModifyAgent** pour configurer le nouvel agent en tant que service Windows. Sinon, si vous incluez le paramètre **-f**, la commande se termine mais génère un avertissement.

Avant de pouvoir exécuter la commande **fteMigrateAgent**, vous devez arrêter l'agent que vous souhaitez migrer à l'aide de la commande **fteStopAgent**.

Si vous exécutez la commande avec le paramètre **-f**, seules les informations relatives à l'agent sont actualisées. Si un fichier requis est manquant, la commande échoue.

En particulier, les fichiers de propriétés, les fichiers XML et les répertoires suivants associés à l'agent sont migrés:

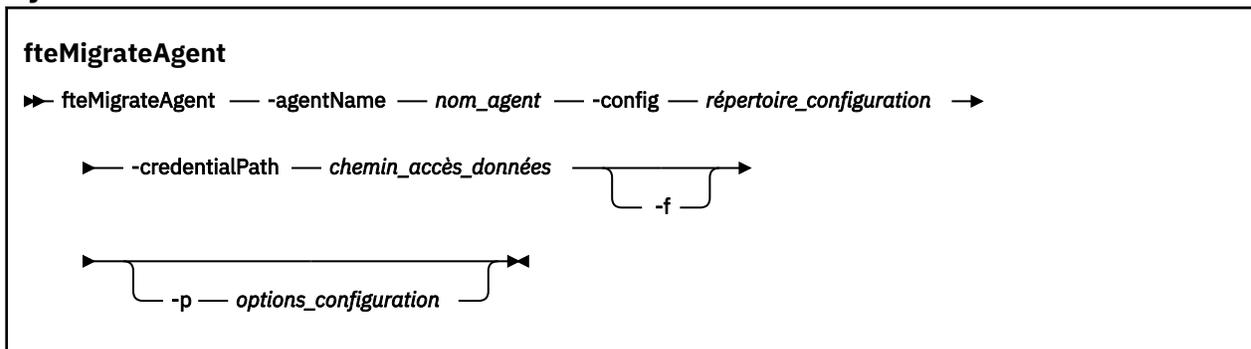
<i>Tableau 37. Fichiers d'agent migrés par la commande d'agent fteMigrate</i>	
Nom du fichier migré par la commande d'agent fteMigrate pour chaque agent	Informations
wmqfte.properties	Le fichier wmqfte.properties est renommé en installation.properties dans IBM WebSphere MQ 7.5 ou version ultérieure.
command.properties	
coordination.properties	
coordination_queue_manager.mqsc	
agent_name_create.mqsc	
agent_name_delete.mqsc	

Tableau 37. Fichiers d'agent migrés par la commande d'agent *fteMigrate* (suite)

Nom du fichier migré par la commande d'agent <i>fteMigrate</i> pour chaque agent	Informations
exits répertoire	La commande copie tous les fichiers du répertoire <i>exits</i> .
S'applique uniquement aux agents standard:	
UserSandboxes.xml	
S'applique uniquement aux agents de pont Connect:Direct :	
ConnectDirectCredentials.xml	
ConnectDirectNodeProperties.xml	
ConnectDirectProcessDefinitions.xml	
S'applique aux agents de pont de protocole uniquement:	
ProtocolBridgeCredentials.xml	
ProtocolBridgeProperties.xml	Ce fichier existe sous IBM WebSphere MQ File Transfer Edition 7.0.4 ou version ultérieure uniquement.

La commande **fteMigrateAgent** migre les fichiers des gestionnaires de files d'attente d'installation, de coordination et de commandes et les copie dans IBM WebSphere MQ 7.5 ou version ultérieure si les fichiers n'existent pas déjà sur IBM WebSphere MQ 7.5 ou version ultérieure. Si les fichiers existent déjà, ils ne sont pas copiés dans le cadre de la commande.

Syntaxe



Paramètres

-agentName *nom_agent*

Obligatoire. Nom de l'agent que vous souhaitez migrer vers IBM WebSphere MQ 7.5 ou version ultérieure.

-config *répertoire_configuration*

Obligatoire. Chemin d'accès au répertoire de configuration de l'installation à partir de laquelle vous migrez l'agent. Exemple: C:\Documents and Settings\All Users\Application Data\IBM\WMQFTE\config

-credentialPath *credentials_path*

Obligatoire. Définit l'emplacement vers lequel les données d'identification doivent être migrées. Ce paramètre peut être un chemin de répertoire dans lequel sont présents des fichiers de données d'identification existants ou un nouvel emplacement pour recevoir un nouveau fichier de données

d'identification. Pour les plateformes z/OS , il peut s'agir d'un fichier partitionné étendu (PDSE) préexistant, avec des membres existants à mettre à jour ou sans membres existants pour inclure un nouveau membre pour ces données d'identification.

Remarque : Si un ensemble de données partitionnées étendu est utilisé, il doit être bloqué.

-f

Facultatif. Force la migration de l'agent même si certains des fichiers de configuration qui sont généralement migrés sont en conflit avec la configuration existante. Par exemple, s'il existe une non-concordance entre les fichiers de propriétés sous Managed File Transfer et les fichiers de propriétés sous IBM WebSphere MQ 7.5 ou version ultérieure, la spécification du paramètre **-f** signifie que cette non-concordance est ignorée.

-p options_configuration

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour localiser la configuration à migrer. Utilisez le nom d'un ensemble d'options de configuration comme valeur du paramètre **-p** . Par convention, il s'agit du nom d'un gestionnaire de files d'attente de coordination. Si vous ne spécifiez pas ce paramètre, l'ensemble d'options de configuration par défaut est utilisé. Pour plus d'informations, voir la section «Options de configuration de MFT sur Multiplatforms», à la page 58.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemples

Dans cet exemple, AGENT3 et sa configuration dans /var/ibm/WMQFTE/config sont migrés vers IBM WebSphere MQ 7.5 ou une version ultérieure:

```
fteMigrateAgent -agentName AGENT3 -config /var/ibm/WMQFTE/config -credentialPath /home/user1/AGENT3
```

Dans cet exemple, tous les agents et leurs configurations dans C:\Documents and Settings\All Users\Application Data\IBM\WMQFTE\config sont migrés vers IBM WebSphere MQ 7.5 ou une version ultérieure. Le chemin d'accès au fichier Windows est placé entre guillemets (""). Le paramètre **-f** est spécifié pour forcer la migration et ignorer les non-concordances de fichier de propriétés:

```
fteMigrateAgent -agentName "*" -config "C:\Documents and Settings\All Users\Application Data\IBM\WMQFTE\config" -credentialPath "C:\Documents and Settings\user1\AGENT3" -p "configurationOption" -f
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Pour plus d'informations sur les codes retour, voir «Codes retour pour MFT», à la page 347.

fteMigrateConfigurationOptions: migration d'une configuration FTE 7.0 vers MQ 7.5 ou version ultérieure

La commande **fteMigrateConfigurationOptions** migre un ensemble d'options de configuration depuis IBM WebSphere MQ File Transfer Edition 7.0 et les copie dans IBM WebSphere MQ 7.5 ou version ultérieure, à condition que les fichiers n'existent pas déjà sur la version cible. Si les fichiers existent déjà, un message s'affiche et la commande ne se poursuit pas.

Remarque : Si vous migrez depuis IBM WebSphere MQ File Transfer Edition 7.0 et que vous souhaitez continuer à utiliser la variable d'environnement FTE_CONFIG, vous pouvez le faire sans modifier la valeur de FTE_CONFIG. Vous pouvez effectuer une migration standard, mais BFG_DATA ne doit pas être défini et FTE_CONFIG doit être défini comme utilisé dans IBM WebSphere MQ File Transfer Edition 7.0.

Important : **ULW** Sous IBM MQ for UNIX, Linux, and Windows, seuls les utilisateurs qui sont des administrateurs d'IBM MQ (et membres du groupe mqm) peuvent exécuter cette commande. Si vous tentez d'exécuter cette commande en tant qu'utilisateur qui n'est pas un administrateur IBM MQ, le message d'erreur BFGCL0502E: Vous n'êtes pas autorisé à effectuer l'opération demandée. et la commande n'est pas exécutée.

z/OS Sur les systèmes z/OS, l'utilisateur doit remplir (au moins) l'une des conditions suivantes pour pouvoir exécuter la commande migrate :

- Etre membre du groupe mqm (si le groupe mqm est défini sur le système).
- **V9.0.0.1** Etre membre du groupe désigné dans la variable d'environnement BFG_GROUP_NAME (si un tel groupe a été désigné).
- **V9.0.0.1** Ne pas avoir de valeur définie dans la variable d'environnement BFG_GROUP_NAME lorsque la commande est exécutée.

Syntaxe

fteMigrateConfigurationOptions

```
► fteMigrateConfigurationOptions — -config — répertoire_configuration — -credentialPath ►  
  
► chemin_accès_données — -configurationOptionsName — nom_options_configuration ►
```

Paramètres

-config (répertoire_configuration)

Obligatoire. Chemin d'accès au répertoire de configuration de l'installation à partir de laquelle vous effectuez la migration. Exemple: C:\Documents and Settings\All Users\Application Data\IBM\WMQFTE\config

-credentialPath (chemin_données)

Obligatoire. Définit l'emplacement vers lequel les données d'identification doivent être migrées. Ce paramètre peut être un chemin de répertoire dans lequel sont présents des fichiers de données d'identification existants ou un nouvel emplacement pour recevoir un nouveau fichier de données d'identification.

Ce paramètre est utilisé pour la migration des propriétés de mot de passe du magasin de clés SSL/TLS et du magasin de clés de confiance présents dans les fichiers `agent.properties`, `coordination.properties` et `command.properties` à partir d'une version du produit antérieure à IBM WebSphere MQ 7.5 vers 7.5 ou ultérieure.

Avant IBM WebSphere MQ 7.5, la propriété de mot de passe, par exemple **coordinationSslTrustStorePassword**, était présente dans le fichier `coordination.properties`, mais a été déplacée dans le fichier `MQMFTCcredentials.xml` dans IBM WebSphere MQ 7.5. Pour déplacer la propriété de mot de passe du fichier `coordination.properties` vers le fichier `MQMFTCcredentials.xml`, **-credentialsPath** est utilisé. Lorsque cette option est utilisée, la commande **fteMigrateConfigurationOptions** recherche dans le fichier `coordination.properties` "coordinationSslTrustStorePassword" et, si elle est présente, migre la propriété vers le fichier `MQMFTCcredentials.xml`.

z/OS Pour les plateformes z/OS, il peut s'agir d'un fichier partitionné étendu (PDSE) préexistant, avec des membres existants à mettre à jour ou sans membres existants pour inclure un nouveau membre pour ces données d'identification.

Remarque : Si un ensemble de données partitionnées étendu est utilisé, il doit s'agir d'un bloc de variables.

-configurationOptionsNom (*configuration_options_name*)

Obligatoire. Nom de l'ensemble d'options de configuration que vous souhaitez migrer. Vous pouvez migrer plusieurs ensembles d'options de configuration à l'aide de l'astérisque (*) pour représenter zéro ou plusieurs caractères. Vous pouvez utiliser un astérisque avec une chaîne. Par exemple, pour migrer tous les ensembles d'options de configuration dont les noms commencent par IBM, utilisez ce paramètre comme suit: -configurationOptionsName IBM*.

Exemples

Dans cet exemple, toutes les configurations du répertoire C:\Documents and Settings\All Users\Application Data\IBM\WMQFTE\config sont migrées. Le chemin de répertoire est placé entre guillemets:

```
fteMigrateConfigurationOptions -config "C:\Documents and Settings\All Users\Application Data\IBM\WMQFTE\config" -credentialPath "C:\Documents and Settings\user1\configurationoptions" -configurationOptionsName *
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Référence associée

«Remarques sur la migration pour MFT», à la page 13

Si vous prévoyez de migrer Managed File Transfer, consultez les informations suivantes.

«Agent fteMigrate: migration d'un agent FTE 7.0 vers MQ 7.5 ou version ultérieure», à la page 494

Si vous souhaitez migrer un agent existant et sa configuration associée depuis IBM WebSphere MQ File Transfer Edition 7.0 vers IBM WebSphere MQ 7.5 ou une version ultérieure, utilisez la commande **fteMigrateAgent** pour effectuer la migration. Cette commande peut être utilisée pour migrer un agent standard, un agent Connect:Direct ou un agent de pont de protocole. La commande peut également être utilisée pour migrer plusieurs agents en une seule demande.

«fteMigrateLogger: migration d'un consignateur de base de données FTE 7.0 vers MQ 7.5 ou version ultérieure», à la page 499

Si vous souhaitez migrer la configuration d'un consignateur de base de données autonome existant depuis IBM WebSphere MQ File Transfer Edition 7.0.1 ou une version ultérieure vers IBM WebSphere MQ 7.5 ou une version ultérieure, utilisez la commande **fteMigrateLogger**.

fteMigrateLogger: migration d'un consignateur de base de données FTE 7.0 vers MQ 7.5 ou version ultérieure

Si vous souhaitez migrer la configuration d'un consignateur de base de données autonome existant depuis IBM WebSphere MQ File Transfer Edition 7.0.1 ou une version ultérieure vers IBM WebSphere MQ 7.5 ou une version ultérieure, utilisez la commande **fteMigrateLogger**.

Vous ne pouvez pas utiliser cette commande pour migrer un consignateur de base de données Java EE : utilisez à la place les informations de la rubrique [Migration d'un consignateur de base de données Java EE](#).

Remarque : Si vous migrez depuis IBM WebSphere MQ 7.0 ou une version ultérieure et que vous souhaitez continuer à utiliser la variable d'environnement FTE_CONFIG, vous pouvez le faire sans modifier la valeur de FTE_CONFIG. Vous pouvez effectuer une migration standard, mais BFG_DATA ne doit pas être défini et FTE_CONFIG doit être défini comme utilisé dans IBM WebSphere MQ 7.0.

Important :  Sous IBM MQ for UNIX, Linux, and Windows, seuls les utilisateurs qui sont des administrateurs d'IBM MQ (et membres du groupe mqm) peuvent exécuter cette commande. Si vous tentez d'exécuter cette commande en tant qu'utilisateur qui n'est pas un administrateur IBM MQ ,

le message d'erreur BFGCL0502E: Vous n'êtes pas autorisé à effectuer l'opération demandée. et la commande n'est pas exécutée.

z/OS Sur les systèmes z/OS, l'utilisateur doit remplir (au moins) l'une des conditions suivantes pour pouvoir exécuter la commande migrate :

- Etre membre du groupe mqm (si le groupe mqm est défini sur le système).
- **V9.0.0.1** Etre membre du groupe désigné dans la variable d'environnement BFG_GROUP_NAME (si un tel groupe a été désigné).
- **V9.0.0.1** Ne pas avoir de valeur définie dans la variable d'environnement BFG_GROUP_NAME lorsque la commande est exécutée.

Si vous avez configuré un consigneur de base de données autonome pour qu'il s'exécute en tant que service Windows , vous ne pouvez pas migrer la configuration de ce consigneur à l'aide de la commande **fteMigrateLogger** . Si vous exécutez la commande **fteMigrateLogger** sur un consigneur configuré pour s'exécuter en tant que service Windows , la commande génère une erreur et ne se poursuit pas. Sinon, si vous incluez le paramètre **-f**, la commande se termine mais génère un avertissement.

Avant d'exécuter la commande **fteMigrateLogger** , arrêtez le consigneur de base de données dont vous souhaitez migrer la configuration sous IBM WebSphere MQ File Transfer Edition 7.0.

Si vous exécutez la commande avec le paramètre **-f**, seules les informations relatives au consigneur sont actualisées. Si un fichier requis est manquant, la commande échoue. En particulier, les fichiers de propriétés suivants et le fichier `.mqsc` associé à la configuration du consigneur sont migrés:

Nom du fichier migré par la commande fteMigrateLogger	Informations
wmqfte.properties	Le fichier wmqfte.properties est basé sur installation.properties dans IBM WebSphere MQ 7.5 ou version ultérieure
command.properties	
coordination.properties	
coordination_queue_manager.mqsc	
databaselogger.properties ou tout autre fichier de propriétés spécifié à l'aide du paramètre -file	databaselogger.properties est utilisé pour créer le fichier logger.properties dans IBM WebSphere MQ 7.5 ou version ultérieure.

La commande **fteMigrateLogger** migre les fichiers des gestionnaires de files d'attente d'installation, de coordination et de commandes et les copie dans IBM WebSphere MQ 7.5 ou version ultérieure à condition que les fichiers n'existent pas déjà sur la version cible. Si les fichiers existent déjà, ils ne sont pas copiés dans le cadre de la commande.

Syntaxe

fteMigrateLogger

```
► fteMigrateLogger — -loggerName — nom_journal — -config — répertoire_configuration →  
  
    ► -credentialPath — chemin_accès_données — -file fichier_propriétés — -f —  
  
    ► -p — options_configuration —
```

Paramètres

-loggerName *nom_connexion*

Obligatoire. Nom que vous souhaitez attribuer à la configuration de consignateur migrée dans IBM WebSphere MQ 7.5 ou version ultérieure. Pour plus d'informations sur les noms de consignateur, qui sont nouveaux pour IBM WebSphere MQ 7.5, voir [logger_name parameter](#).

-config *répertoire_configuration*

Obligatoire. Chemin d'accès au répertoire de configuration de l'installation à partir de laquelle la configuration du consignateur est migrée.

-credentialPath *credentials_path*

Obligatoire. Définit l'emplacement vers lequel les données d'identification doivent être migrées. Ce paramètre peut être un chemin de répertoire dans lequel sont présents des fichiers de données d'identification existants ou un nouvel emplacement pour recevoir un nouveau fichier de données d'identification. Pour les plateformes z/OS, il peut s'agir d'un fichier partitionné étendu (PDSE) préexistant, avec des membres existants à mettre à jour ou sans membres existants pour inclure un nouveau membre pour ces données d'identification.

Remarque : Si un ensemble de données partitionnées étendu est utilisé, il doit s'agir d'un bloc de variables.

-file *fichier_propriétés*

Facultatif. Indique le fichier de propriétés du consignateur de base de données à migrer. Ce paramètre est requis uniquement si le fichier de propriétés n'utilise pas le nom et le chemin d'accès par défaut suivants: *configuration_directory/coordination_qmgr_name/databaselogger.properties*

-f

Facultatif. Force la migration même si certains des fichiers de configuration qui sont généralement migrés sont en conflit avec la configuration existante. Par exemple, s'il existe une non-concordance entre les fichiers de propriétés du consignateur de base de données sous IBM WebSphere MQ File Transfer Edition et les fichiers de propriétés sous IBM WebSphere MQ 7.5 ou version ultérieure, la spécification du paramètre **-f** signifie que cette non-concordance est ignorée.

-p *options_configuration*

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour localiser la configuration du consignateur à migrer. Utilisez le nom d'un ensemble d'options de configuration comme valeur du paramètre **-p**. Par convention, il s'agit du nom d'un gestionnaire de files d'attente de coordination. Si vous ne spécifiez pas ce paramètre, l'ensemble d'options de configuration par défaut est utilisé. Pour plus d'informations, voir la section «Options de configuration de MFT sur Multiplatforms», à la page 58.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, la configuration d'un consigneur de base de données autonome situé dans `/var/ibm/WMQFTE/config` est migrée vers IBM WebSphere MQ 7.5 et se nomme FTELOGGER1:

```
fteMigrateLogger -loggerName FTELOGGER1 -config /var/ibm/WMQFTE/config  
-credentialPath /home/user1/FTELOGGER1
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Pour plus d'informations sur les codes retour, voir [«Codes retour pour MFT»](#), à la page 347.

Après l'exécution de la commande `fteMigrateLogger`

Pour vérifier la migration, une fois que vous avez exécuté la commande `fteMigrateLogger`, démarrez le consigneur de base de données dont vous avez migré la configuration sous IBM WebSphere MQ 7.5 ou une version ultérieure à l'aide de la commande [«fteStartLogger»](#), à la page 542 .

Référence associée

[«Remarques sur la migration pour MFT»](#), à la page 13

Si vous prévoyez de migrer Managed File Transfer, consultez les informations suivantes.

[«Agent fteMigrate: migration d'un agent FTE 7.0 vers MQ 7.5 ou version ultérieure»](#), à la page 494

Si vous souhaitez migrer un agent existant et sa configuration associée depuis IBM WebSphere MQ File Transfer Edition 7.0 vers IBM WebSphere MQ 7.5 ou une version ultérieure, utilisez la commande `fteMigrateAgent` pour effectuer la migration. Cette commande peut être utilisée pour migrer un agent standard, un agent Connect:Direct ou un agent de pont de protocole. La commande peut également être utilisée pour migrer plusieurs agents en une seule demande.

[«fteMigrateConfigurationOptions: migration d'une configuration FTE 7.0 vers MQ 7.5 ou version ultérieure»](#), à la page 497

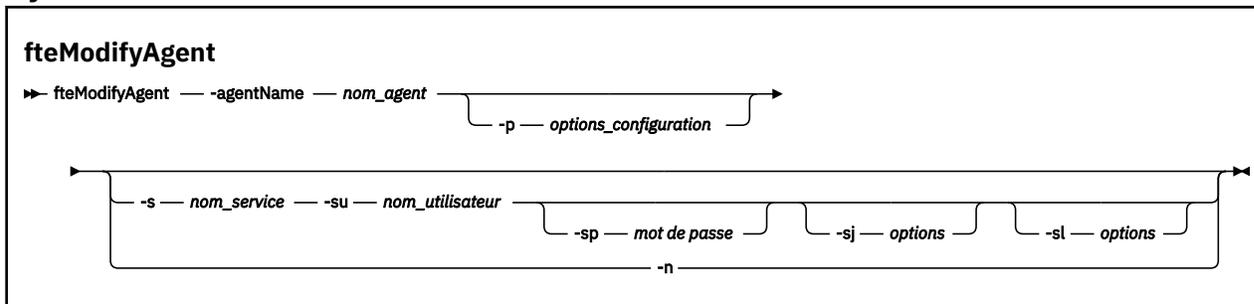
La commande `fteMigrateConfigurationOptions` migre un ensemble d'options de configuration depuis IBM WebSphere MQ File Transfer Edition 7.0 et les copie dans IBM WebSphere MQ 7.5 ou version ultérieure, à condition que les fichiers n'existent pas déjà sur la version cible. Si les fichiers existent déjà, un message s'affiche et la commande ne se poursuit pas.

Windows

Agent `fteModify`(exécutez un agent MFT en tant que service Windows)

La commande `fteModifyAgent` modifie un agent existant afin qu'il puisse être exécuté en tant que service Windows . Cette commande est disponible uniquement sous Windows et doit être exécutée par un utilisateur qui est un administrateur IBM MQ et un membre du groupe `mqm`.

Syntaxe



Paramètres

-agentName *nom_agent*

Obligatoire. Nom de l'agent à modifier.

-p *options_configuration*

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour modifier l'agent. Par convention, utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande **fteModifyAgent** utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Spécifiez le paramètre facultatif **-p** uniquement si vous souhaitez utiliser des options de configuration différentes de vos valeurs par défaut. Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

-s *nom_service*

Facultatif. Indique que l'agent doit être exécuté en tant que service Windows . Si vous ne spécifiez pas *service_name*, le service est nommé mqmftAgentAGENTQMGR, où *AGENT* est le nom de l'agent et *QMGR* est le nom de votre gestionnaire de files d'attente d'agent.

Le nom d'affichage du service, qui est affiché dans la fenêtre Windows **Services** de la colonne **Nom** , est toujours **Managed File Transfer Agent AGENT@QMGR**.

Remarque : Si l'agent redistribuable doit s'exécuter en tant que service Windows, vous devez définir la variable d'environnement **BFG_DATA** dans l'environnement système pour que le service fonctionne.

-su *nom_utilisateur*

Facultatif. Lorsque l'agent doit être exécuté en tant que service Windows , ce paramètre indique le nom du compte sous lequel le service doit être exécuté. Pour exécuter l'agent à l'aide d'un compte utilisateur de domaine Windows , indiquez la valeur dans le formulaire DomainName\UserName. Pour exécuter le service à l'aide d'un compte à partir du domaine intégré local, spécifiez la valeur dans le formulaire UserName.

Le compte utilisateur Windows que vous spécifiez à l'aide du paramètre **-su** doit disposer du droit **Log on as a service** . Pour plus d'informations sur l'octroi de ce droit, voir [Conseils pour l'exécution d'un agent ou d'un consignateur MFT en tant que service Windows](#).

Ce paramètre est obligatoire lorsque **-s** est spécifié.

-sp *mot_de_passe*

Facultatif. Ce paramètre est valide uniquement lorsque **-s** est spécifié.

-sj *options*

Facultatif. Lorsque l'agent est démarré en tant que service Windows , ce paramètre définit une liste d'options sous la forme -D ou -X qui sera transmise à la machine virtuelle Java (JVM). Les options sont séparées par le signe dièse (#) ou le point-virgule (;). Si vous avez besoin d'incorporer un # ou ; , placez-les entre apostrophes.

Ce paramètre est valide uniquement lorsque **-s** est spécifié.

Pour plus d'informations sur la façon dont la commande **fteModifyAgent** gère la validation des mises à jour des options JVM, voir [Conseils pour la mise à jour des options JVM de l'agent ou du consignateur](#).

-sl *options*

Facultatif. Définit le niveau de journalisation du service Windows . Les options valides sont: error, info, warn, debug. La valeur par défaut est info. Cette option peut être utile si vous rencontrez des problèmes avec le service Windows . Le fait de le définir sur debug fournit des informations plus détaillées dans le fichier journal du service.

Ce paramètre est valide uniquement lorsque **-s** est spécifié.

-n

Facultatif. Indique que l'agent doit être exécuté en tant que processus normal. Cette option est mutuellement exclusive avec l'option **-s**. Si ni l'option **-s** ni l'option **-n** n'est spécifiée, l'agent est configuré en tant que processus Windows normal.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, AGENT1 est modifié pour s'exécuter en tant que service Windows :

```
fteModifyAgent -agentName AGENT1 -s -su fteuser -sp ftepassword
```

Dans cet exemple, AGENT1 est modifié pour supprimer le service Windows :

```
fteModifyAgent -agentName AGENT1
```

Vous devez arrêter l'agent à modifier à l'aide de la commande [fteStopAgent](#) pour pouvoir exécuter la commande d'agent fteModify.

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Référence associée

«Agent fteCreate(créer un agent MFT)», à la page 399

La commande **fteCreateAgent** crée un Managed File Transfer Agent et sa configuration associée.

«fteModifyLogger (exécutez un consignateur MFT en tant que service Windows)», à la page 504

La commande **fteModifyLogger** permet de modifier un consignateur Managed File Transfer afin qu'il puisse être exécuté en tant que service Windows. Vous pouvez utiliser cette commande uniquement sur les plateformes Windows, vous devez être exécuté par un utilisateur qui est un administrateur IBM MQ et un membre du groupe mqm, et vous devez d'abord arrêter le consignateur à l'aide de la commande **fteStopLogger**.

Information associée

[Démarrage d'un agent MFT en tant que service Windows](#)

[Conseils pour l'exécution d'un agent ou d'un consignateur MFT en tant que service Windows](#)

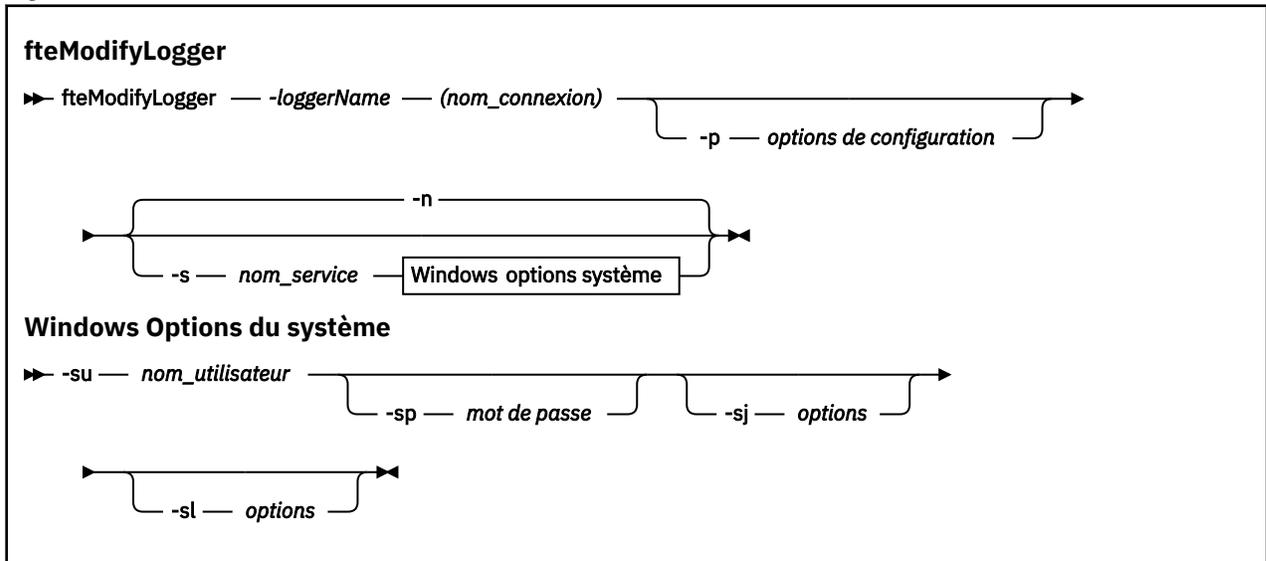
Windows fteModifyLogger (exécutez un consignateur MFT en tant que service Windows)

La commande **fteModifyLogger** permet de modifier un consignateur Managed File Transfer afin qu'il puisse être exécuté en tant que service Windows. Vous pouvez utiliser cette commande uniquement sur les plateformes Windows, vous devez être exécuté par un utilisateur qui est un administrateur IBM MQ et un membre du groupe mqm, et vous devez d'abord arrêter le consignateur à l'aide de la commande **fteStopLogger**.

Objet

Un consignateur autonome, qu'il s'agisse d'un fichier ou d'une base de données, est affiché en tant que consignateur "Managed File Transfer pour l'ensemble de propriétés *nom_consignateur@logger_qm*" dans la colonne **Nom** de l'application **Services**. La valeur de *logger_qm* est le nom du gestionnaire de files d'attente de commandes du consignateur.

Syntaxe



Paramètres

-loggerName (nom_journal)

Obligatoire. Nom du consignateur Managed File Transfer à modifier.

-p options_configuration

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour modifier le consignateur. Par convention, utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande **fteModifyLogger** utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Spécifiez le paramètre facultatif **-p** uniquement si vous souhaitez utiliser des options de configuration différentes de vos valeurs par défaut. Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

-s nom_service

Facultatif. Indique que le consignateur doit être exécuté en tant que service Windows. Si vous ne spécifiez pas `service_name`, le service est nommé `mqmftLogger` ou `LOGGERQMGR`, où `LOGGER` est le nom du consignateur et `QMGR` est le nom de votre gestionnaire de files d'attente de consignateur.

Le nom d'affichage du service, qui est affiché dans la fenêtre **Services** Windows dans la colonne **Nom**, est toujours **Managed File Transfer Logger `LOGGER@QMGR`**.

-su nom_utilisateur

Obligatoire lorsque **-s** est spécifié. Indique le nom du compte sous lequel le service Windows doit s'exécuter. Pour exécuter l'agent à l'aide d'un compte utilisateur de domaine Windows, spécifiez la valeur dans le formulaire `DomainName\UserName`. Pour exécuter le service à l'aide d'un compte à partir du domaine intégré local, spécifiez la valeur dans le formulaire `UserName`.

Le compte utilisateur Windows que vous spécifiez à l'aide du paramètre **-su** doit avoir le droit de se connecter en tant que service. Pour plus d'informations sur l'octroi de ce droit, voir [Conseils pour l'exécution d'un agent ou d'un consignateur MFT en tant que service Windows](#).

-sp mot_de_passe

Facultatif. Valide uniquement lorsque **-s** est spécifié. Mot de passe du compte utilisateur défini par le paramètre **-su**.

Si vous ne spécifiez pas ce paramètre lorsque vous spécifiez le paramètre **-s**, vous êtes averti que vous devez définir le mot de passe à l'aide de l'outil Windows Services pour que le service puisse démarrer correctement.

-sj options

Facultatif. Valide uniquement lorsque **-s** est spécifié. Lorsque le consignateur est démarré en tant que service Windows, ce paramètre définit une liste d'options sous la forme **-D** ou **-X** qui sera transmise à la machine virtuelle Java. Les options sont séparées par le signe dièse (**#**) ou le point-virgule (**;**). Si vous avez besoin d'incorporer un **#** ou **;**, placez-les entre apostrophes (**'**).

Pour plus d'informations sur la façon dont la commande **fteModifyLogger** gère la validation des mises à jour des options JVM, voir [Conseils pour la mise à jour des options JVM de l'agent ou du consignateur](#).

-sl options

Facultatif. Valide uniquement lorsque **-s** est spécifié. Définit le niveau de journalisation du service Windows. Les options valides sont: **error**, **info**, **warn**, **debug**. La valeur par défaut est **info**. Cette option peut être utile si vous rencontrez des problèmes avec le service Windows. Le fait de le définir sur **debug** fournit des informations plus détaillées dans le fichier journal du service.

-n

Facultatif. Indique que le consignateur doit être exécuté en tant que processus normal. Cette option est mutuellement exclusive avec l'option **-s**. Si ni l'option **-s** ni l'option **-n** n'est spécifiée, le consignateur est configuré en tant que processus Windows normal.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Vous devez arrêter le consignateur à l'aide de la commande [fteStopLogger](#), avant d'exécuter la commande **fteModifyLogger**.

Dans cet exemple, un consignateur nommé **logger1** a déjà été créé. Cette commande montre comment le consignateur peut être modifié pour s'exécuter en tant que service Windows :

```
fteModifyLogger -loggerName logger1 -s -su fteuser -sp ftepassword
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Référence associée

[«fteStartLogger»](#), à la page 542

La commande **fteStartLogger** démarre une application de journalisation Managed File Transfer.

[«fteStopLogger»](#), à la page 546

La commande **fteStopLogger** arrête un consignateur Managed File Transfer.

Information associée

[Démarrage d'un agent MFT en tant que service Windows](#)

[Conseils pour l'exécution d'un agent ou d'un consignateur MFT en tant que service Windows](#)

fteObfuscate: chiffrement des données sensibles

La commande **fteObfuscate** chiffre les données sensibles dans les fichiers de données d'identification. Ainsi, le contenu des fichiers de données d'identification n'est plus lu par une personne qui a accès au fichier.

Objet

Les propriétés de nom d'utilisateur et de mot de passe dans les fichiers de données d'identification peuvent être brouillées. Ces propriétés sont transformées en une nouvelle propriété associée, avec un suffixe de chiffrement. Exemple :

```
<!--
  MQMFTCredentials properties before
-->
<tns:logger name="logger1" user="user1" password="passw0rd"/>
<tns:file path="$HOME/trust.jks" password="passw0rd"/>

<!--
  MQMFTCredentials properties after
-->
<tns:logger name="logger1" userCipher="e71vKCg2pf" passwordCipher="se71vKCg"/>
<tns:file path="$HOME/trust.jks" passwordCipher="e71vKCg2pf"/>

<!--
  ProtocolBridgeCredentials Properties before
-->
<tns:user name="Fred" serverUserId="fred" serverPassword="passw0rd"/>

<!--
  ProtocolBridgeCredentials properties after
-->
<tns:user name="Fred" serverUserIdCipher="e51vVCg2pf" serverPasswordCipher="se51vBCg"/>

<!--
  ConnectDirectCredentials properties before
-->
<tns:user name="fteuser" ignorecase="true" pattern="wildcard"
  cdUserId="cdUser" cdPassword="cdPassword" pNodeUserId="pnodeUser"
  pNodePassword="pnodePassword">
  <tns:snode name="snode1" pattern="wildcard" userId="snodeUser" password="snodePassword"/>
</tns:user>

<!--
  ConnectDirectCredentials properties after
-->
<tns:user name="fteuser" ignorecase="true" pattern="wildcard"
  cdUserIdCipher="e71vKCg2pf" cdPasswordCipher="se71vKCg"
  pNodeUserIdCipher="2f1vgCg6df" pNodePasswordCipher="e71vKCg2pf">
  <tns:snode name="snode1" pattern="wildcard" userIdCipher="e51vVCg2pf" passwordCipher="se51vBCg"/>
</tns:user>
```

Syntaxe

fteObfuscate

► fteObfuscate — -credentialsFile — *nom_fichier_données d'identification* ►►

Paramètre

-credentialsFile

Obligatoire. Nom du fichier de données d'identification dont le contenu sera brouillé.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, le contenu `MQMFTCcredentials.xml` est brouillé.

```
fteObfuscate -credentialsFile /home/fteuser/MQMFTCcredentials.xml
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Référence associée

«Authentification de connexion MFT et IBM MQ», à la page 42

L'authentification de connexion permet à un gestionnaire de files d'attente d'être configuré pour authentifier les applications à l'aide d'un ID utilisateur et d'un mot de passe fournis. Si la sécurité est activée pour le gestionnaire de files d'attente associé et que les données d'identification (ID utilisateur et mot de passe) sont requises, la fonction d'authentification de connexion doit être activée pour que la connexion à un gestionnaire de files d'attente puisse être établie. L'authentification de connexion peut être exécutée en mode compatibilité ou en mode d'authentification MQCSP.

«Format de fichier de données d'identification MFT», à la page 835

Le fichier `MQMFTCcredentials.xml` contient les informations d'ID utilisateur et de mot de passe sensibles. Les éléments du fichier `MQMFTCcredentials.xml` doivent être conformes au schéma `MQMFTCcredentials.xsd`. La sécurité des fichiers de données d'identification est de la responsabilité de l'utilisateur.

Agent ftePing: vérifiez si un agent MFT est actif

La commande **ftePingAgent** envoie une commande ping à un agent Managed File Transfer pour déterminer si l'agent est accessible et, dans l'affirmative, s'il est en mesure de répondre à une requête simple.

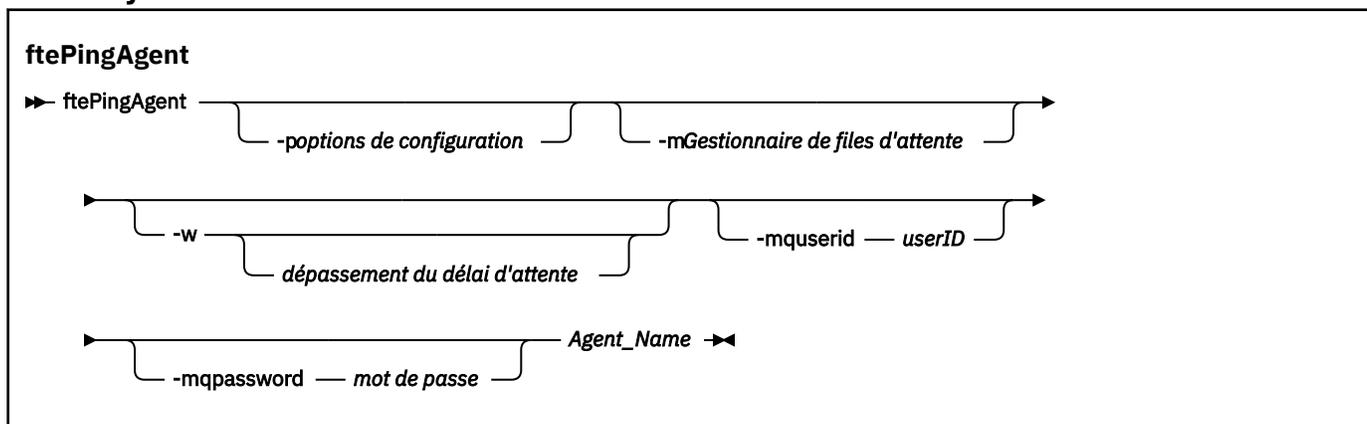
Objet

Utilisez la commande **ftePingAgent** pour vérifier si un agent Managed File Transfer est accessible et, si tel est le cas, s'il est en mesure de répondre à une requête simple sur les lignes `are you there?`. Voici un exemple de sortie de cette commande:

```
C:\> ftePingAgent AGENT86
5724-H72 Copyright IBM Corp. 2008, 2023. TOUS DROITS RÉSERVÉS
BFGPR0127W: Aucun fichier de données d'identification n'a été spécifié pour la connexion à IBM
MQ. Le système considère donc que l'authentification IBM MQ a été désactivée.
BFGCL0212I: Exécution d'une demande ping à l'agent AGENT86
BFGCL0213I: AGENT86 a répondu à la commande ping en 0.094 secondes.
```

Spécifiez le paramètre facultatif **-p** pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de votre ensemble par défaut. Pour plus d'informations, voir [Options de configuration](#).

Syntaxe



Paramètres

-p (options de configuration)

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour émettre la demande à un agent. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut. Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé. Pour plus d'informations, voir [Options de configuration](#).

-m (gestionnaire de files d'attente)

Facultatif. Nom du gestionnaire de files d'attente auquel l'agent à contacter (ping) est connecté. Si vous ne spécifiez pas le paramètre -m, le gestionnaire de files d'attente utilisé est déterminé à partir de l'ensemble des options de configuration utilisées.

-w (délai d'attente)

Facultatif. Indique que la commande doit attendre jusqu'à *timeout* secondes pour que l'agent réponde. Si vous ne spécifiez pas de délai d'attente ou si vous spécifiez une valeur de délai d'attente de -1, la commande attend indéfiniment jusqu'à ce que l'agent réponde. Si vous ne spécifiez pas cette option, le délai d'attente de réponse par défaut de l'agent est de cinq secondes.

Si *timeout* a été spécifié, les messages de commande **ftePingAgent** expirent après le double de la valeur de *timeout* au lieu d'être placés dans la file d'attente de rebut désignée. Les messages de commande n'ont pas de délai d'attente si la commande a été définie pour attendre indéfiniment.

-mquserid (ID utilisateur)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de files d'attente de commandes.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de commandes. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

(nom de l'agent)

Obligatoire. Nom de l'agent Managed File Transfer auquel vous souhaitez envoyer une commande ping.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, la commande exécute une commande ping sur l'agent AGENT1, qui est connecté à QM_MERCURY. La commande attend jusqu'à 40 secondes que AGENT1 réponde avant de revenir.

```
ftePingAgent -m QM_MERCURY -w 40 AGENT1
```

Codes retour

0

La commande a abouti. L'agent est actif et peut traiter les transferts.

1

La commande a échoué. La commande n'a pas pu envoyer de message à l'agent.

2

La commande s'est terminée avec un délai d'attente. La commande a envoyé un message à l'agent, mais celui-ci n'a pas répondu dans le délai imparti.

Référence associée

[«Agents fteList: répertorie les agents MFT pour un gestionnaire de files d'attente de coordination», à la page 484](#)

La commande **fteListAgents** permet de répertorier tous les agents Managed File Transfer enregistrés auprès d'un gestionnaire de files d'attente de coordination particulier.

[«fteShowAgentDetails», à la page 530](#)

La commande **fteShowAgentDetails** permet d'afficher les détails d'un Managed File Transfer Agent particulier. Il s'agit des détails stockés par le gestionnaire de files d'attente de coordination Managed File Transfer de l'agent.

[«Que faire si vous pensez que votre transfert de fichiers est bloqué», à la page 321](#)

Sur un système très chargé ou en cas de problèmes de réseau entre les agents source et cible, les transferts peuvent parfois sembler bloqués dans un état de file d'attente ou de récupération. Un certain nombre de facteurs peuvent être à l'origine de cette situation.

fteRAS: collecte des informations de traitement des incidents liés à MFT

La commande **fteRAS** collecte des informations de traitement des incidents (donnéesMustGather) pour Managed File Transfer. Les informations collectées par **fteRAS** sont spécifiques à l'installation de Managed File Transfer sur le système sur lequel le programme est exécuté.

Objet

Utilisez la commande **fteRAS** pour exécuter l'outil de collecte d'informations sur la fiabilité, la disponibilité et la facilité de maintenance (RAS) si vous avez besoin de collecter des informations de traitement des incidents à utiliser pour trouver une solution lorsqu'un agent Managed File Transfer, un consignateur de base de données ou une autre commande signale un problème ou un échec de fonctionnement correct.

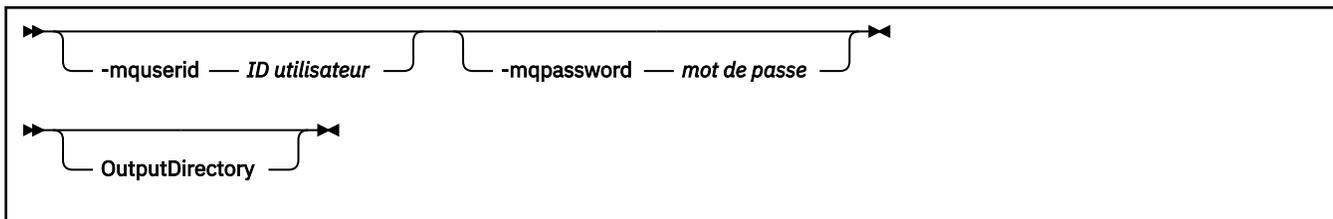
Lorsque vous exécutez la commande **fteRAS**, le répertoire de sortie dans lequel le fichier archive résultant (.zip) est placé peut être l'emplacement par défaut ou un répertoire de votre choix.

Syntaxe

fteRAS

► **fteRAS** *-p options_configuration* *-l PDSLibraryName* ◀

Sécurité



Paramètres

-p options_configuration

Facultatif. Détermine l'ensemble des options de configuration utilisées pour collecter les informations d'identification et de résolution des problèmes, par exemple, la liste des agents. Utilisez le nom d'un ensemble d'options de configuration comme valeur pour le paramètre **-p**. Par convention, ce nom est le nom d'un gestionnaire de file d'attente de coordination. Si vous ne spécifiez pas ce paramètre, l'ensemble d'options de configuration par défaut est utilisé.

z/OS -l

Facultatif. z/OS seulement. Indique le nom d'une bibliothèque PDS contenant des scripts JCL qui appellent des commandes MQMFT pour un agent ou un consignateur particulier. Cette option est toujours définie lorsque la commande est exécutée à partir du script JCL BFGRAS d'une bibliothèque PDS de commandes, de sorte que tous les membres de la bibliothèque PDS soient capturés dans le répertoire de sortie.

-mquserid ID utilisateur

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de files d'attente de commandes.

-mqpassword mot_de_passe

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de commandes. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid** mais ne spécifiez pas également **-mqpassword**, vous êtes invité à fournir le mot de passe associé. Le mot de passe ne s'affiche pas à l'écran.

OutputDirectory

Facultatif. Répertoire à utiliser lorsque vous collectez les données RAS et que le fichier de sortie, par exemple `fteRAS.zip`, est stocké après la collecte des données. Si le répertoire n'existe pas, il est créé. L'emplacement par défaut est le répertoire des journaux `mqft`.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemples

Linux **UNIX** Sous UNIX et Linux, pour stocker le fichier de sortie `fteRAS.zip` dans le répertoire `/var/mqm/errors`, exécutez **fteRAS** comme illustré dans l'exemple suivant:

```
fteRAS /var/mqm/errors
```

Le message suivant confirme que la commande a abouti:

```
BFGCL0604I: fteRAS a abouti. La sortie est stockée dans /var/mqm/errors/fteRAS.zip
```

Windows Sous Windows, pour stocker le fichier de sortie `fteRAS.zip` dans le répertoire des erreurs par défaut pour une nouvelle installation de IBM MQ, exécutez **fteRAS** comme illustré dans l'exemple suivant:

```
fteRAS "C:\ProgramData\IBM\MQ\errors"
```

Le message suivant confirme que la commande a abouti:

```
BFGCL0604I: fteRAS a abouti. La sortie est stockée dans C:\ProgramData\IBM\MQ\errors\fteRAS.zip
```

Remarque : Pour IBM MQ 8.0 ou version ultérieure, s'il ne s'agit pas d'une nouvelle installation de cette version du produit, l'emplacement du répertoire des erreurs peut être différent sur votre système. Pour plus d'informations, voir [Emplacement du répertoire de programme et des données sous Windows](#).

IBM i Sous IBM i, pour copier le fichier de sortie dans /QIBM/UserData/mqm/errors, exécutez la commande **fteRAS** à partir de Qshell, comme illustré dans l'exemple suivant:

```
/QIBM/ProdData/mqm/bin/fteRAS /QIBM/UserData/mqm/errors
```

Le message suivant confirme que la commande a abouti:

```
BFGCL0604I: fteRAS a abouti. La sortie est stockée dans /QIBM/UserData/mqm/errors/fteRAS.zip
```

Référence associée

«traitement des incidentsMFT», à la page 277

Utilisez les informations de référence suivantes pour vous aider à diagnostiquer les erreurs dans Managed File Transfer:

V 9.0.0 fteSetAgentLogLevel: Définir le niveau de journalisation de l'agent de pont de protocole MFT

La commande **fteSetAgentLogLevel** permet d'activer ou de désactiver la consignation des informations de diagnostic pour les transferts de fichiers entre un agent de pont de protocole Managed File Transfer et des serveurs de fichiers FTP/SFTP/FTPS.

Objet

Cette commande permet d'activer ou de désactiver la consignation pour la collecte des informations de diagnostic sur le transfert de fichiers entre un agent de pont de protocole Managed File Transfer et des serveurs de fichiers FTP/SFTP/FTPS. La journalisation de l'agent de pont de protocole Managed File Transfer est plus simple, distincte de, mais peut être exécutée en parallèle avec la trace.

Les commandes envoyées par l'agent de pont de protocole aux serveurs de fichiers FTP/SFTP/FTPS et les réponses reçues sont consignées dans un fichier journal pour un diagnostic simple des problèmes pouvant se produire lors des transferts de fichiers.

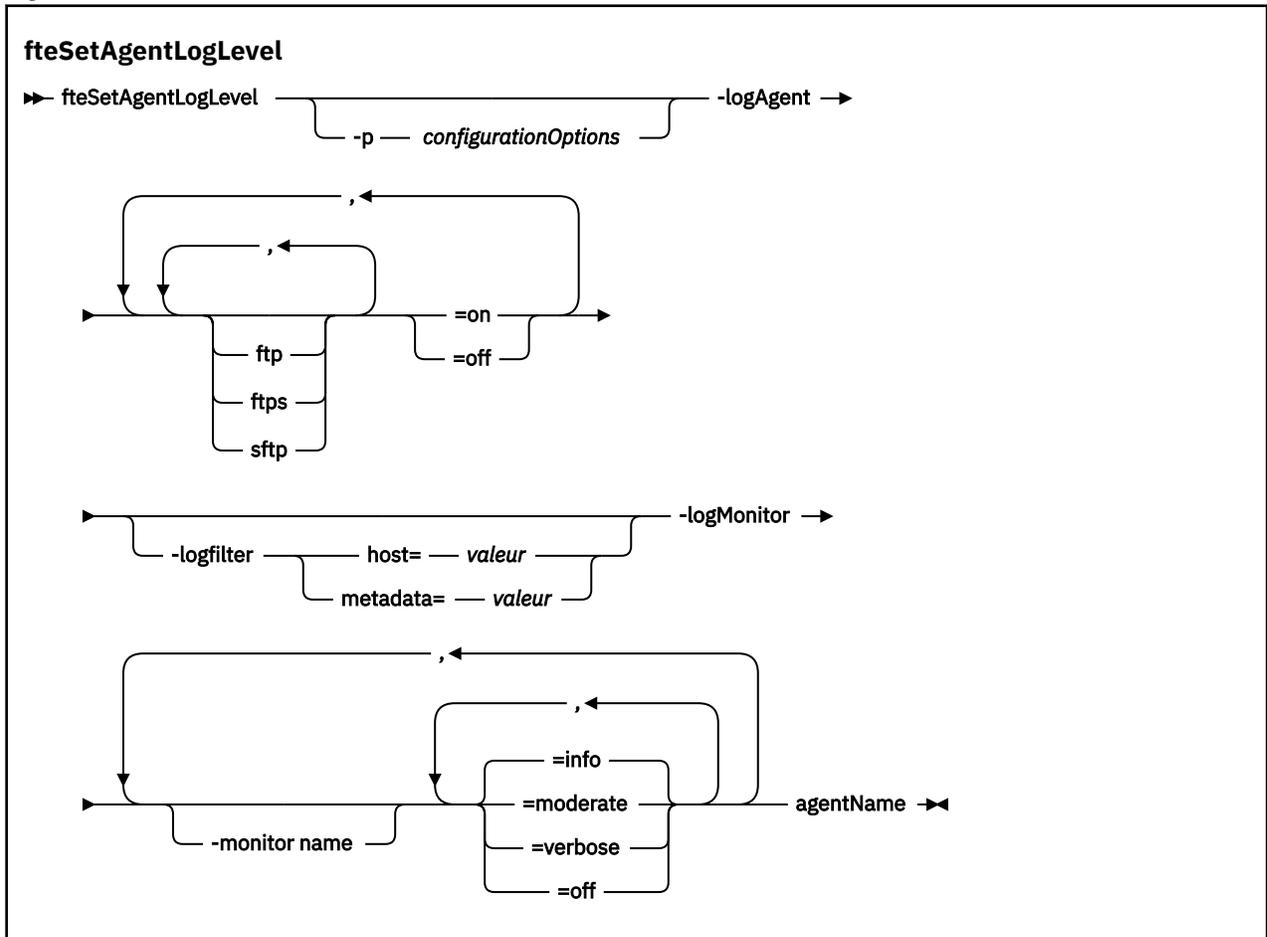
Vous pouvez activer, désactiver et définir le niveau de journalisation dont vous avez besoin de deux manières:

- Utilisez la commande **fteSetAgentLogLevel** pour activer ou désactiver la journalisation lorsque l'agent de pont de protocole est en cours d'exécution. Vous n'avez pas besoin de redémarrer l'agent de pont de protocole pour que la consignation commence.
- Utilisez le fichier `agent.properties` pour activer ou désactiver la journalisation au démarrage de l'agent. Si vous utilisez cette méthode pour activer ou désactiver la journalisation et que l'agent est en cours d'exécution, vous devez redémarrer l'agent pour que la journalisation commence. Pour plus d'informations, voir [Le fichier agent.properties](#).

Une interaction entre un agent de pont de protocole et un serveur de fichiers est traitée comme un événement et chaque événement est écrit dans un fichier `agentevent?.log`. Le caractère '?' dans le nom de journal représente une valeur numérique. Le fichier dont le nom contient 0 (zéro) contient l'ensemble de messages capturés le plus récent. Le fichier journal est stocké dans le dossier `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/nom_agent/`.

Vous pouvez définir d'autres propriétés de journalisation, par exemple, pour limiter la taille du fichier journal et le nombre de fichiers journaux conservés, dans le [fichier agent.properties](#).

Syntaxe



Paramètres

-logAgent component=operation

Obligatoire.

V 9.0.3

Important : Vous ne devez sélectionner qu'une seule des options **logAgent** et **logMonitor**. Si vous tentez de sélectionner les deux options, vous recevez un message d'erreur.

Nom de l'agent de pont de protocole pour lequel la consignation est activée ou désactivée. La journalisation de l'agent de pont de protocole peut être activée ou désactivée pour trois composants: les serveurs *ftp*, *ftpset* *sftp* . Indiquez l'un des trois types de serveur possibles et ajoutez une valeur d'opération pour désactiver la journalisation ou activer pour l'agent de pont de protocole auquel ils se connectent.

V 9.0.3

Si un composant commence par un signe plus (+), la liste des composants qui suit le signe plus est ajoutée à tout composant de journal existant actuellement consigné.

Les options d'opération de niveau de journalisation valides sont les suivantes:

désactivé

Désactivez toutes les consignations pour un agent de pont de protocole. Il s'agit de l'option par défaut.

```
fteSetAgentLogLevel -logAgent =off PBA1
```

Pour désactiver la consignation pour un composant spécifié auquel l'agent de pont de protocole se connecte, utilisez les commandes suivantes:

```
fteSetAgentLogLevel -logAgent ftp=off PBA1
```

```
fteSetAgentLogLevel -logAgent ftps=off PBA1
```

```
fteSetAgentLogLevel -logAgent sftp=off PBA1
```

activé

Pour activer la consignation pour les trois composants de serveur de fichiers possibles auxquels l'agent de pont de protocole se connecte, utilisez la commande suivante:

```
fteSetAgentLogLevel -logAgent =on PBA1
```

Pour activer la consignation pour un composant spécifié auquel un agent de pont de protocole se connecte, utilisez les commandes suivantes:

```
fteSetAgentLogLevel -logAgent ftp=on PBA1
```

```
fteSetAgentLogLevel -logAgent ftps=on PBA1
```

```
fteSetAgentLogLevel -logAgent sftp=on PBA1
```

Pour plus d'options de configuration, voir [«Exemple 1»](#), à la page 516 et [«Exemple 2»](#), à la page 516.

-agentName

Obligatoire. Nom de l'agent de pont de protocole pour lequel la consignation est activée ou désactivée.

-logFilter *filter=valeur*

Facultatif. Permet de limiter la consignation aux critères de filtrage spécifiés. Vous devez spécifier une valeur pour **host** ou pour le transfert de fichiers spécifié par l'utilisateur **metadata**.

host

Indiquez l'une des trois valeurs de filtrage suivantes: nom d'hôte, liste de noms d'hôte ou d'adresses IP séparés par des virgules. Pour consigner les commandes et les réponses du serveur FTP `ftpprod.ibm.com`, utilisez la commande suivante:

```
fteSetAgentLogLevel -logAgent ftp=on -logFilter host=ftpprod.ibm.com PBA1
```

Pour consigner les commandes et les réponses de tous les serveurs SFTP dont les adresses IP commencent par `9.182.*`, utilisez la commande suivante:

```
fteSetAgentLogLevel -logAgent sftp=on -logFilter host=9.182.* PBA1
```

métadonnées

Indiquez tout texte, tel que défini par l'utilisateur lors de la création du transfert, au format *clé=valeur*. Par exemple, **metadata**="BANK=WORLD BANK". Pour activer la consignation pour tout serveur FTP qui se connecte à l'agent de pont de protocole PBA1 et filtrer les transferts

de fichiers afin de les inclure uniquement s'ils contiennent la clé **metadata** et la valeur "BANK=WORLD BANK", utilisez la commande suivante:

```
fteSetAgentLogLevel -logAgent ftp=on metadata="BANK=WORLD BANK" PBA1
```

Remarque : Pour filtrer par métadonnées, la valeur que vous filtrez doit être spécifiée sous le paramètre **-md** dans le cadre d'un transfert de fichier. Pour plus d'informations, voir la section «[fteCreateTransfer: démarrer un nouveau transfert de fichier](#)», à la page 447.

V 9.0.3 **-logMonitor monitor name=niveau de journalisation**

Obligatoire.

Important : Vous ne devez sélectionner qu'une seule des options **logAgent** et **logMonitor**. Si vous tentez de sélectionner les deux options, vous recevez un message d'erreur.

Liste de moniteurs de ressources séparés par des virgules pour activer ou désactiver la journalisation, où:

Nom du moniteur

Facultatif. Nom du moniteur de ressources, ou liste de noms, pour activer ou désactiver la consignation. Si vous n'indiquez pas de nom, la consignation est activée pour tous les moniteurs de ressources de l'agent.

niveau de journalisation

Obligatoire. L'une des entrées suivantes :

info

Activez la consignation du niveau d'information. Il s'agit de la valeur par défaut.

Pour activer la journalisation de niveau `info` pour le moniteur MON1 de l'agent AGENT1:

```
fteSetAgentLogLevel -logMonitor MON1=info AGENT1
```

modéré

Activez la consignation de niveau moyen.

Pour activer la journalisation de niveau `moderate` pour les moniteurs MON1 et MON2 de l'agent AGENT1:

```
fteSetAgentLogLevel -logMonitor MON1, MON2=moderate AGENT1
```

prolix

Pour activer la consignation de niveau prolix.

Activez la journalisation de niveau `verbose` pour tous les moniteurs de l'agent AGENT1:

```
fteSetAgentLogLevel -logMonitor =verbose AGENT1
```

désactivé

Désactivez la fonction de consignation.

Pour désactiver la consignation pour les moniteurs MON1 et MON2 de l'agent AGENT1:

```
fteSetAgentLogLevel -logMonitor MON1, MON2=off AGENT1
```

Pour désactiver la consignation pour le moniteur MON1 et activer la `info` consignation de niveau pour le moniteur MON2 de l'agent AGENT1:

```
fteSetAgentLogLevel -logMonitor MON1=off, MON2=info AGENT1
```

Pour désactiver la consignation pour tous les moniteurs de l'agent AGENT1:

```
fteSetAgentLogLevel -logMonitor =off AGENT1
```

V 9.0.3 -p configurationOptions

Facultatif. Détermine l'ensemble des options de configuration utilisées pour définir le niveau de journalisation de l'agent. Utilisez le nom d'un ensemble d'options de configuration comme valeur pour le paramètre **-p**.

Par convention, il s'agit du nom d'un gestionnaire de files d'attente de coordination. Si vous ne spécifiez pas ce paramètre, l'ensemble d'options de configuration par défaut est utilisé.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

V 9.0.3 Si le même nom de moniteur de ressources est répété dans une commande, la dernière occurrence d'un composant est considérée comme valide. Exemple :

```
fteSetAgentLogLevel -logMonitor MON1=info, MONZ=off, MON1=off AGENT1
```

désactive la consignation pour le moniteur de ressources MON1.

V 9.0.3



Attention : Si vous avez indiqué des noms de moniteur de ressources inexistant dans la commande, aucune erreur ne s'affiche sur la console.

Exemple 1

Dans cet exemple, plusieurs composants sont spécifiés dans une commande, à l'aide d'un groupe délimité par des commandes. La journalisation est activée pour les serveurs FTP et SFTP, mais désactivée pour les serveurs FTPS auxquels l'agent de pont de protocole PBA1 se connecte.

```
fteSetAgentLogLevel -logAgent ftp=on,ftps=off,sftp=on PBA1
```

Vous pouvez également séparer les composants avec une virgule pour obtenir le même résultat, par exemple ;

```
fteSetAgentLogLevel -logAgent ftp,sftp=on,ftps=off PBA1
```

Exemple 2

Dans cet exemple, le même composant est répété dans une commande. La dernière instance d'une paire *component=operation* est considérée comme valide. Cet exemple désactive la journalisation des serveurs FTP auxquels l'agent de pont de protocole PBA1 se connecte.

```
fteSetAgentLogLevel -logAgent ftp=on,ftp=off PBA1
```

L'exemple précédent a le même effet que cet exemple:

```
fteSetAgentLogLevel -logAgent ftp=off PBA1
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Concepts associés

«Le pont de protocole», à la page 219

Le pont de protocole permet à votre réseau Managed File Transfer (MFT) d'accéder aux fichiers stockés sur un serveur de fichiers en dehors de votre réseau MFT, dans votre domaine local ou à un emplacement distant. Ce serveur de fichiers peut utiliser les protocoles de réseau FTP, FTPS ou SFTP. Chaque serveur de fichiers a besoin d'au moins un agent dédié. L'agent dédié est appelé agent de pont de protocole. Un agent de pont peut interagir avec plusieurs serveurs de fichiers.

Référence associée

«Le fichier MFT agent.properties», à la page 564

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.

«`fteCreateTransfer`: démarrer un nouveau transfert de fichier», à la page 447

La commande **`fteCreateTransfer`** crée et démarre un nouveau transfert de fichier à partir de la ligne de commande. Cette commande peut démarrer un transfert de fichier immédiatement, planifier un transfert de fichier à une heure et une date ultérieures, répéter un transfert planifié une ou plusieurs fois et déclencher un transfert de fichiers sous certaines conditions.

fteSetAgentTraceLevel

La commande **`fteSetAgentTraceLevel`** permet de modifier le niveau de trace en cours d'un agent de manière dynamique.

Objet

Utilisez cette commande pour activer et désactiver la trace d'agent ou pour modifier le niveau de trace d'agent défini. Lorsque vous utilisez la commande **`fteSetAgentTraceLevel`**, vous n'avez pas besoin d'arrêter et de redémarrer un agent pour modifier le niveau de trace. Les fichiers de trace générés se trouvent dans `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name/logs/trace%PID%/trace%PID%.txt`, où `%PID%` est l'ID de processus de l'instance d'agent.



Avertissement :

Multi Lorsque vous utilisez IBM WebSphere MQ 7.5 ou une version ultérieure sous Multiplateformes, seul l'utilisateur sur lequel s'exécute le processus d'agent peut exécuter la commande **`fteSetAgentTraceLevel`**.

V 9.0.2 **z/OS** Pour z/OS, pour Continuous Delivery, à partir de IBM MQ 9.0.2, la commande **`fteSetAgentTraceLevel`** peut être exécutée par:

- Le même ID utilisateur que celui sous lequel le processus d'agent s'exécute.
- Les membres du groupe spécifiés par la propriété d'agent **`adminGroup`**.

V 9.0.0.1 Cela s'applique également à l'édition Long Term Support de IBM MQ 9.0.0 Fix Pack 1.

Pour plus d'informations, voir la propriété **`adminGroup`** dans «Le fichier MFT agent.properties», à la page 564.

Dans la IBM WebSphere MQ 7.5 et les versions ultérieures, la commande **`fteSetAgentTraceLevel`** écrit également une trace pour le contrôleur de processus de l'agent. Les fichiers de trace générés se trouvent dans `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name/logs/pctrace%PID%/pctrace%PID%.txt`, où `%PID%` est l'ID de processus de l'instance d'agent.

Vous pouvez également utiliser la commande pour que le processus d'agent génère un Javacore. L'agent génère un fichier Javacore dans le répertoire suivant: `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name`.

Etant donné que l'exécution de la trace peut affecter vos performances de manière significative et peut produire une grande quantité de données de trace, exécutez la trace avec précaution et uniquement

lorsque cela est nécessaire. En règle générale, activez la fonction de trace uniquement lorsque le technicien de maintenance IBM vous le demande.



Avertissement :

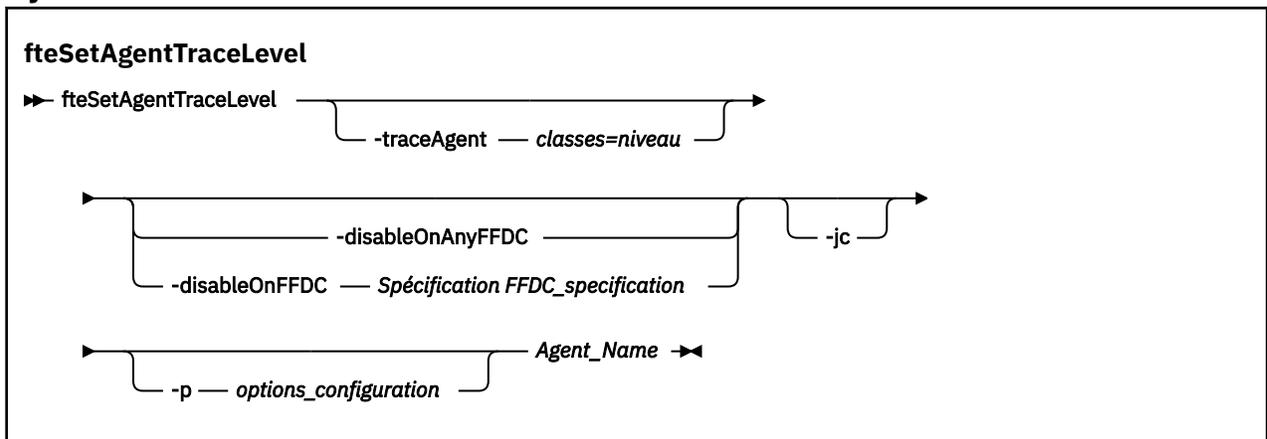
1. Vous devez exécuter cette commande sur le système sur lequel l'agent est en cours d'exécution.
2. Les traces et la consignation ne sont pas conservées lors du redémarrage d'un agent.

Si l'agent s'arrête et est redémarré par le processus du contrôleur de processus, les traces dynamiques et les journaux ne sont pas en vigueur tant que le fichier `agent.properties` n'a pas été mis à jour pour inclure les propriétés de trace et de journal requises.

Vous pouvez définir d'autres propriétés de trace, par exemple la taille du fichier de trace et le nombre de fichiers de trace à conserver, dans le fichier `agent.properties`. Ces propriétés sont décrites dans [Propriétés avancées de l'agent](#).

Spécifiez le paramètre facultatif `-p` pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de votre ensemble par défaut. Pour plus d'informations, voir «Le fichier `MFT.agent.properties`», à la page 564.

Syntaxe



Paramètres

-traceAgent classes=niveau

Obligatoire. Niveau pour définir la trace de l'agent et les classes à appliquer à la trace.

V 9.0.5 Depuis la IBM MQ 9.0.5, vous pouvez spécifier une liste de spécifications de classe séparées par un signe deux-points. Cette option permet de définir le traçage de différentes classes, à différents niveaux. Exemple :

```
fteSetAgentTraceLevel -traceAgent com.ibm.wmqfte.agent=all:com.ibm.wmqfte.cmdhandler=moderate AGENT1
```

Vous pouvez toujours spécifier une liste séparée par des virgules de spécifications de classe auxquelles vous souhaitez que le niveau de trace s'applique. Si vous ne spécifiez pas ce paramètre, le niveau de trace est appliqué à toutes les classes d'agent. Utilisez le format suivant :

```
classes=level
```

Exemple :

```
com.ibm.wmqfte=all
```

Vous pouvez remplacer *classes* par un nom de package Managed File Transfer pour tracer un package spécifique uniquement. Toutefois, comme cette option capture uniquement un sous-ensemble du comportement de l'agent, il n'est généralement pas recommandé d'utiliser le filtrage de package.

Si (*classes*) commencent par un signe plus (+), la liste des classes de trace qui suit le signe plus est ajoutée à toutes les classes de trace existantes actuellement tracées.

Les options de niveau de trace valides sont les suivantes et sont répertoriées par ordre croissant de taille et de détail du fichier de trace:

désactivé

Désactive la trace de l'agent mais continue d'écrire des informations dans les fichiers journaux. Il s'agit de l'option par défaut.

flux

Capture des données pour les points de trace associés au flux de traitement dans l'agent.

modéré

Capture une quantité modérée d'informations de diagnostic dans la trace.

prolix

Capture une quantité prolix d'informations de diagnostic dans la trace.

Tous

Définit la trace d'agent à exécuter sur toutes les classes d'agent.

Pour démarrer le traçage complet de l'agent, exécutez la commande suivante:

```
fteSetAgentTraceLevel -traceAgent =all AGENT_NAME
```

Pour arrêter la fonction de trace complète de l'agent, exécutez la commande suivante:

```
fteSetAgentTraceLevel -traceAgent =off AGENT_NAME
```

-disableOnAnyFFDC

Facultatif. Si ce paramètre est spécifié, la trace est désactivée sur l'agent lorsqu'il génère un fichier FFDC (First Failure Data Capture).

Vous ne pouvez spécifier qu'un seul des paramètres **-disableOnAnyFFDC** et **-disableOnFFDC**.

-disableOnFFDC Spécification FFDC

Facultatif. Si ce paramètre est spécifié, la trace est désactivée sur l'agent lorsqu'il génère un fichier FFDC (First Failure Data Capture) qui correspond à la *spécification FFDC_specification*. *FFDC_specification* est une liste de valeurs séparées par des virgules. Le format des valeurs peut être:

nom_classe

Nom de la classe d'origine de l'outil de diagnostic de premier niveau (FFDC). Par exemple, `com.ibm.wmqfte.classA`.

nom_classe:probe_ID

Nom de la classe et ID sonde de l'emplacement de la classe d'où provient l'outil de diagnostic de premier niveau (FFDC). Par exemple, `com.ibm.wmqfte.classB:1`.

Vous ne pouvez spécifier qu'un seul des paramètres **-disableOnAnyFFDC** et **-disableOnFFDC**.

-jc

Facultatif. Demande à l'agent de générer un fichier javacore. L'équipe du service IBM peut vous demander d'exécuter la commande avec ce paramètre pour faciliter le diagnostic des problèmes. Ce paramètre ne peut être utilisé avec aucun autre paramètre à l'exception de **-p**.

-p options_configuration

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour définir le niveau de trace de l'agent. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre

que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

Agent_Name

Obligatoire. Nom du Managed File Transfer Agent pour lequel vous souhaitez définir le niveau de trace.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, le niveau de trace est défini sur `all` pour toutes les classes pour `AGENT1`:

```
fteSetAgentTraceLevel -traceAgent com.ibm.wmqfte=all AGENT1
```

Dans cet exemple, le niveau de trace est défini sur `all` pour les classes `com.ibm.wmqfte.agent.Agent` et `com.ibm.wmqfte.cmdhandler` pour `AGENT1`:

```
fteSetAgentTraceLevel -traceAgent com.ibm.wmqfte.agent.Agent,com.ibm.wmqfte.cmdhandler=moderate AGENT1
```

Dans cet exemple, les sous-classes sont exclues de la trace car le paramètre **-traceLevel** est défini sur `off`. Toutes les classes commençant par `com.ibm.outer` sont tracées au niveau `prolix`, à l'exception des classes commençant par `com.ibm.outer.inner`:

```
fteSetAgentTraceLevel -traceAgent com.ibm.outer=verbose AGENT1  
fteSetAgentTraceLevel -traceAgent +com.ibm.outer.inner=off AGENT1
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

fteSetLoggerTraceLevel

La commande **fteSetLoggerTraceLevel** permet de modifier dynamiquement le niveau de trace en cours d'un consignateur Managed File Transfer .

Objet

Cette commande permet d'activer et de désactiver la trace du consignateur ou de modifier le niveau de trace du consignateur défini. Lorsque vous utilisez la commande **fteSetLoggerTraceLevel** , il n'est pas nécessaire d'arrêter et de redémarrer un consignateur pour modifier le niveau de trace. Les fichiers de trace générés se trouvent dans `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/loggers/logger_name/logs/trace%PID%/trace%PID%.txt`, où `%PID%` est l'ID de processus de l'instance de consignateur.

Dans la IBM WebSphere MQ 7.5 et les versions ultérieures, la commande **fteSetLoggerTraceLevel** écrit également une trace pour le contrôleur de processus du consignateur. Les fichiers de trace générés se trouvent dans `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/loggers/logger_name/logs/pctrace%PID%/pctrace%PID%.txt`, où `%PID%` est l'ID de processus de l'instance de consignateur.

La commande peut également être utilisée pour que le processus du consignateur génère un Javacore. Le consignateur génère un fichier Javacore dans le répertoire suivant: `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/loggers/logger_name`.

Etant donné que l'exécution de la trace peut affecter vos performances de manière significative et peut produire une grande quantité de données de trace, exécutez la trace avec précaution et uniquement lorsque cela est nécessaire. En règle générale, activez la fonction de trace uniquement lorsque le technicien de maintenance IBM vous le demande.



Avertissement :

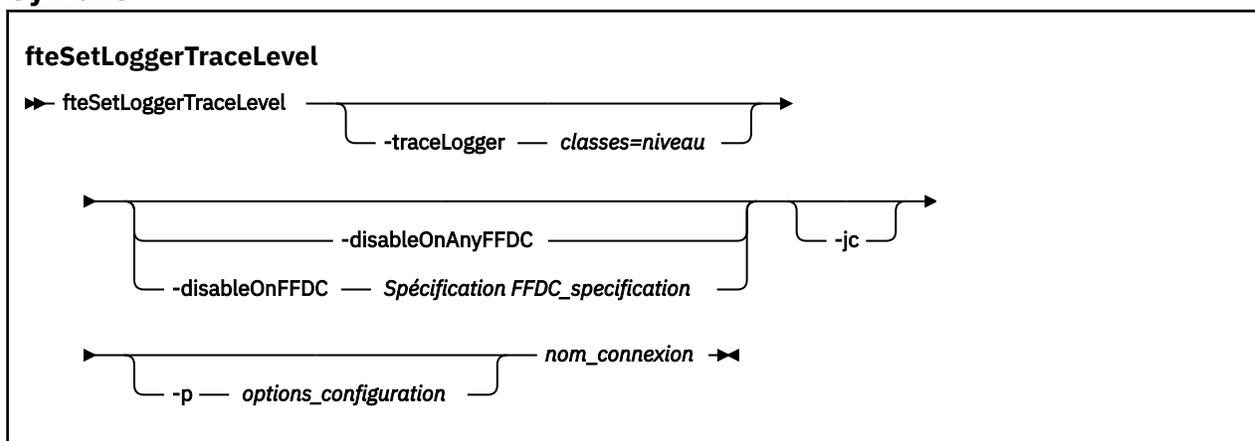
1. Vous devez exécuter cette commande sur le système sur lequel l'agent est en cours d'exécution.
2. Les traces et la consignment ne sont pas conservées lors du redémarrage d'un agent.

Si l'agent s'arrête et est redémarré par le processus du contrôleur de processus, les traces dynamiques et les journaux ne sont pas en vigueur tant que le fichier `agent.properties` n'a pas été mis à jour pour inclure les propriétés de trace et de journal requises.

Vous pouvez définir d'autres propriétés de trace, par exemple la taille du fichier de trace et le nombre de fichiers de trace à conserver, dans le fichier `logger.properties`. Ces propriétés sont décrites dans [Propriétés du consignateur](#).

Spécifiez le paramètre facultatif `-p` pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de votre ensemble par défaut. Pour plus d'informations, voir la section «[Propriétés de configuration du consignateur MFT](#)», à la page 120.

Syntaxe



Paramètres

-traceLogger classes=niveau

Obligatoire. Niveau permettant de définir la trace du consignateur et les classes auxquelles appliquer la trace.

V 9.0.5 Depuis la IBM MQ 9.0.5, vous pouvez spécifier une liste de spécifications de classe séparées par un signe deux-points. Cette option permet de définir le traçage de différentes classes, à différents niveaux. Exemple :

```
fteSetLoggerTraceLevel -traceLogger com.ibm.wmqfte.logger=all:com.ibm.wmqfte.cmdhandler=moderate  
LOGGER1
```

Vous pouvez toujours spécifier une liste séparée par des virgules de spécifications de classe auxquelles vous souhaitez que le niveau de trace s'applique. Si vous ne spécifiez pas ce paramètre, le niveau de trace est appliqué à toutes les classes d'agent. Utilisez le format suivant :

```
classes=level
```

Exemple :

```
com.ibm.wmqfte=all
```

Indiquez une liste séparée par des virgules des spécifications de classe auxquelles vous souhaitez appliquer le niveau de trace. Si vous ne spécifiez pas ce paramètre, le niveau de trace est appliqué à toutes les classes de consignateur.

Si (*classes*) commencent par un signe plus (+), la liste des classes de trace qui suivent le signe plus est ajoutée à toutes les classes de trace existantes actuellement tracées.

Les options de niveau de trace valides sont les suivantes et sont répertoriées par ordre croissant de taille et de détail du fichier de trace:

désactivé

Désactive la trace du consignateur mais continue d'écrire des informations dans les fichiers journaux. Il s'agit de l'option par défaut.

flux

Capture des données pour les points de trace associés au flux de traitement dans le consignateur.

modéré

Capture une quantité modérée d'informations de diagnostic dans la trace.

prolix

Capture une quantité prolix d'informations de diagnostic dans la trace.

Tous

Définit la trace du consignateur à exécuter sur toutes les classes du consignateur.

-disableOnAnyFFDC

Facultatif. Si ce paramètre est spécifié, la trace est désactivée sur le consignateur lorsqu'il génère un fichier FFDC (First Failure Data Capture).

Vous ne pouvez spécifier qu'un seul des paramètres **-disableOnAnyFFDC** et **-disableOnFFDC**.

-disableOnFFDC Spécification FFDC

Facultatif. Si ce paramètre est spécifié, la trace est désactivée sur le consignateur lorsqu'il génère un fichier FFDC (First Failure Data Capture) qui correspond à la *spécification FFDC_specification*. *FFDC_specification* est une liste de valeurs séparées par des virgules. La valeur peut être l'un des formats suivants:

nom_classe

Nom de la classe d'origine de l'outil de diagnostic de premier niveau (FFDC). Par exemple, `com.ibm.wmqfte.classA`.

nom_classe:probe_ID

Nom de la classe et ID sonde de l'emplacement de la classe d'où provient l'outil de diagnostic de premier niveau (FFDC). Par exemple, `com.ibm.wmqfte.classB:1`.

Vous ne pouvez spécifier qu'un seul des paramètres **-disableOnAnyFFDC** et **-disableOnFFDC**.

-jc

Facultatif. Demande au consignateur de générer un fichier javacore. L'équipe du service IBM peut vous demander d'exécuter la commande avec ce paramètre pour faciliter le diagnostic de l'incident. Vous ne pouvez pas utiliser le paramètre **-jc** avec un autre paramètre.

-p options_configuration

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour définir le niveau de trace du consignateur. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble

des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

nom_consignateur

Obligatoire. Nom du Managed File Transfer Logger pour lequel vous souhaitez définir le niveau de trace.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, le niveau de trace est défini sur `all` pour toutes les classes pour `LOGGER1`:

```
fteSetLoggerTraceLevel -traceLogger com.ibm.wmqfte=all LOGGER1
```

Dans cet exemple, le niveau de trace est défini sur `all` pour les classes `com.ibm.wmqfte.logger.logger` et `com.ibm.wmqfte.cmdhandler` pour `LOGGER1`:

```
fteSetLoggerTraceLevel -traceLogger com.ibm.wmqfte.logger.logger,com.ibm.wmqfte.cmdhandler=moderate  
LOGGER1
```

Dans cet exemple, les sous-classes sont exclues de la trace car le paramètre **-traceLevel** est défini sur `off`. Toutes les classes commençant par `com.ibm.outer` sont tracées au niveau `prolix`, à l'exception des classes commençant par `com.ibm.outer.inner`:

```
fteSetLoggerTraceLevel -traceLogger com.ibm.outer=verbose LOGGER1  
fteSetLoggerTraceLevel -traceLogger +com.ibm.outer.inner=off LOGGER1
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

fteSetProductId: définissez l'ID produit d'enregistrement **z/OS SCRT**

fteSetProductId est utilisé pour définir le type de produit par rapport auquel l'utilisation de Managed File Transfer doit être enregistrée pour l'installation. Cette commande est valide uniquement sous z/OS.

Objet

Cette commande peut être exécutée à tout moment, une fois qu'au moins un gestionnaire de files d'attente de coordination a été défini ou que le fichier [installation.properties](#), pour l'installation, a été créé.

Pour plus d'informations sur l'enregistrement de l'utilisation du produit, voir [Génération de rapports sur les informations produit](#).

Syntaxe



Paramètres

Type de produit pour l'enregistrement de l'utilisation:

Indiquez l'une des valeurs suivantes :

MFT

L'utilisation est enregistrée en tant que produit Managed File Transfer autonome, avec l'ID produit 5655-MF9.

ADVANCED

L'utilisation est enregistrée dans le cadre d'un produit IBM MQ Advanced for z/OS , avec l'ID produit 5655-AV9.

ADVANCEDVUE

L'utilisation est enregistrée dans le cadre d'un produit IBM MQ Advanced for z/OS, Value Unit Edition , avec l'ID produit 5655-AV1.

Codes retour

0

La commande a abouti.

1

La commande a échoué ou si le type de produit n'a pas été défini sur la valeur demandée.

Tâches associées

«Configuration du gestionnaire de files d'attente de coordination pour MFT», à la page 99

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.

Commandes fteSetup: créez le fichier MFT command.properties

La commande **fteSetupCommands** crée le fichier Managed File Transfer `command.properties` . Ce fichier de propriétés spécifie les détails du gestionnaire de files d'attente qui se connecte au réseau IBM MQ lorsque vous émettez des commandes.

Important :  Sous IBM MQ for UNIX, Linux, and Windows, seuls les utilisateurs qui sont des administrateurs d'IBM MQ (et membres du groupe mqm) peuvent exécuter cette commande. Si vous tentez d'exécuter cette commande en tant qu'utilisateur qui n'est pas un administrateur IBM MQ , le message d'erreur BFGCL0502E: Vous n'êtes pas autorisé à effectuer l'opération demandée . et la commande n'est pas exécutée.

 Sur les systèmes z/OS, l'utilisateur doit remplir (au moins) l'une des conditions suivantes pour pouvoir exécuter la commande `migrate` :

- Etre membre du groupe mqm (si le groupe mqm est défini sur le système).
-  Etre membre du groupe désigné dans la variable d'environnement `BFG_GROUP_NAME` (si un tel groupe a été désigné).

-CanalconnectionQMgr(connection_qmgr_channel)

Facultatif. Nom de canal utilisé pour la connexion au gestionnaire de files d'attente de connexion. Si vous spécifiez le paramètre **-connectionQMgrChannel**, vous devez également spécifier le paramètre **-connectionQMgrHost**.

-p (options_configuration)

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour configurer un gestionnaire de files d'attente de commandes. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande **fteSetupCommands** utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

-credentialsFile (filePath)

Facultatif. Chemin d'accès complet à un fichier de données d'identification existant ou nouveau, dans lequel les détails de l'authentification IBM MQ sont ajoutés.

Cette commande prend en charge l'ajout d'un ensemble de détails d'authentification IBM MQ à un fichier de données d'identification Managed File Transfer nommé. Utilisez-la si l'authentification de connexion IBM MQ a été activée. Si vous mettez à jour les détails existants, vous devez utiliser le paramètre force **-f**.

-userid (nom_utilisateur)

Facultatif. ID utilisateur indiqué pour associer les données d'identification détaillées. Si vous ne spécifiez pas d'ID utilisateur, les données d'identification détaillées sont valables pour tous les utilisateurs. Vous devez aussi spécifier le paramètre **-credentialsFile**.

-f

Facultatif. Force l'écrasement du fichier `command.properties` existant avec les détails spécifiés dans cette commande.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

```
fteSetupCommands -connectionQMgr QM_NEPTUNE -connectionQMgrHost 9.146.157.241  
-connectionQMgrPort 1414 -connectionQMgrChannel SYSTEM.DEF.SVRCONN
```

Codes retour**0**

La commande a abouti.

1

La commande a échoué.

Référence associée

«[Le fichier MFT command.properties](#)», à la page 559

Le fichier `command.properties` indique le gestionnaire de files d'attente de commandes auquel vous devez vous connecter lorsque vous émettez des commandes et les informations dont Managed File Transfer a besoin pour contacter ce gestionnaire de files d'attente.

«[fteSetupCoordination](#)», à la page 527

La commande **fteSetupCoordination** crée des fichiers de propriétés ainsi que le répertoire du gestionnaire de file d'attente de coordination pour Managed File Transfer.

fteSetupCoordination

La commande **fteSetupCoordination** crée des fichiers de propriétés ainsi que le répertoire du gestionnaire de file d'attente de coordination pour Managed File Transfer.

Important : **ULW** Sous IBM MQ for UNIX, Linux, and Windows, seuls les utilisateurs qui sont des administrateurs d'IBM MQ (et membres du groupe mqm) peuvent exécuter cette commande. Si vous tentez d'exécuter cette commande en tant qu'utilisateur qui n'est pas un administrateur IBM MQ, le message d'erreur BFGCL0502E: Vous n'êtes pas autorisé à effectuer l'opération demandée. et la commande n'est pas exécutée.

z/OS Sur les systèmes z/OS, l'utilisateur doit remplir (au moins) l'une des conditions suivantes pour pouvoir exécuter la commande migrate :

- Etre membre du groupe mqm (si le groupe mqm est défini sur le système).
- **V9.0.0.1** Etre membre du groupe désigné dans la variable d'environnement BFG_GROUP_NAME (si un tel groupe a été désigné).
- **V9.0.0.1** Ne pas avoir de valeur définie dans la variable d'environnement BFG_GROUP_NAME lorsque la commande est exécutée.

Objet

Utilisez la commande **fteSetupCoordination** pour créer les objets Managed File Transfer suivants :

- Répertoire du gestionnaire de file d'attente de coordination
- Répertoire de données mqft (s'il n'existe pas)
- Un fichier installation.properties
- Un fichier coordination.properties

Cette commande fournit également les commandes MQSC suivantes que vous devez exécuter dans votre gestionnaire de file d'attente de coordination afin de configurer Managed File Transfer. Les commandes MQSC créent une rubrique, une chaîne de rubrique, la file d'attente SYSTEM.FTE et les files d'attente de consignateur de base de données par défaut. Elles mettent également à jour une liste de noms et définissent la valeur ENABLED pour l'attribut PSMODE du gestionnaire de file d'attente de coordination.

z/OS Si le gestionnaire de file d'attente de coordination se trouve sous z/OS, avant d'exécuter ces commandes MQSC, vous devez vous assurer que les objets requis suivants existent :

- La file d'attente SYSTEM.BROKER.DEFAULT.STREAM
- La liste de noms SYSTEM.QPUBSUB.QUEUE.NAMELIST
- Les flux SYSTEM.BROKER.DEFAULT.STREAM et SYSTEM.BROKER.ADMIN.STREAM

```
DEFINE TOPIC('SYSTEM.FTE') TOPICSTR('SYSTEM.FTE') REPLACE
ALTER TOPIC('SYSTEM.FTE') NPMSGDLV(ALLAVAIL) PMSGDLV(ALLAVAIL)
DEFINE QLOCAL(SYSTEM.FTE) LIKE(SYSTEM.BROKER.DEFAULT.STREAM) REPLACE
ALTER QLOCAL(SYSTEM.FTE) DESCR('Stream for WMQFTE 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)
```

Pour plus d'informations sur les fichiers de propriétés, voir [Options de configuration](#).

z/OS Si vous utilisez z/OS, vous pouvez émettre la commande **fteSetupCoordination** et d'autres commandes JCL avec des scripts générés depuis le jeu de données de bibliothèque PDSE du

-userid (nom_utilisateur)

Facultatif. ID utilisateur indiqué pour associer les données d'identification détaillées. Si vous ne spécifiez pas d'ID utilisateur, les données d'identification détaillées sont valables pour tous les utilisateurs. Vous devez aussi spécifier le paramètre **-credentialsFile**.

-f

Facultatif. Force le remplacement de la configuration de gestionnaire de file d'attente de coordination existante par les détails spécifiés dans cette commande.

-default

Facultatif. Met à jour les options de configuration par défaut avec les options associées au gestionnaire de file d'attente de coordination spécifié dans cette commande.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, les objets requis sont configurés pour un gestionnaire de file d'attente de coordination appelé QM_SATURN, auquel la connexion est établie en mode client :

```
fteSetupCoordination -coordinationQMgr QM_SATURN  
-coordinationQMgrHost myhost.ibm.com -coordinationQMgrPort 1415  
-coordinationQMgrChannel SYSTEM.DEF.SVRCONN
```

Codes retour**0**

La commande a abouti.

1

La commande a échoué.

Concepts associés

[«Options de configuration de MFT sur Multiplatforms»](#), à la page 58

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.

[«Configuration de MQMFTCredentials.xml sur z/OS»](#), à la page 71

Si Managed File Transfer (MFT) est configuré avec la sécurité activée, l'authentification de connexion requiert tous les agents MFT, ainsi que les commandes qui se connectent à un gestionnaire de files d'attente, pour fournir les données d'identification par ID utilisateur et mot de passe.

Tâches associées

[«Configuration du gestionnaire de files d'attente de coordination pour MFT»](#), à la page 99

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

[«Le fichier MFT agent.properties»](#), à la page 564

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.

Information associée

[«Propriétés SSL du fichier coordination.properties»](#), à la page 624

fteShowAgentDetails

La commande **fteShowAgentDetails** permet d'afficher les détails d'un Managed File Transfer Agent particulier. Il s'agit des détails stockés par le gestionnaire de files d'attente de coordination Managed File Transfer de l'agent.

Objet

Vous pouvez exécuter la commande **fteShowAgentDetails** à partir de n'importe quel système pouvant se connecter au gestionnaire de file d'attente de coordination. Cette commande utilise le fichier `coordination.properties` pour se connecter au gestionnaire de files d'attente de coordination.

Spécifiez le paramètre facultatif **-p** pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de votre ensemble par défaut. Pour plus d'informations, voir la section «Options de configuration de MFT sur Multiplatforms», à la page 58.

Les informations d'état de l'agent générées par cette commande sont générées à partir des messages d'état publiés par l'agent dans `SYSTEM.FTE`. Ces messages sont décrits dans «Format de message d'état de l'agent MFT», à la page 757. Les informations de statut générées par la commande **fteShowAgentDetails** indiquent le statut de l'agent au moment de la publication du dernier message de statut. La fréquence de ces messages d'état dépend de la valeur de la propriété **agentStatusPublishRateLimit**. Pour plus d'informations, voir «Le fichier MFT `agent.properties`», à la page 564.

Pour IBM WebSphere MQ 7.5.0 Fix Pack 1 ou version ultérieure, spécifiez le paramètre facultatif **-d** pour cette commande si vous souhaitez afficher les informations de diagnostic relatives à un agent local. Ces informations incluent les transferts en cours, les transferts planifiés, les moniteurs et les profondeurs de file d'attente d'agent. Vous pouvez utiliser ces informations pour déterminer la santé et l'état d'un agent local.

 Pour z/OS, depuis IBM MQ 9.0.2 et IBM MQ 9.0.0 Fix Pack 1, le paramètre **-d** ne peut être spécifié que si la commande **fteShowAgentDetails** est exécutée par:

- Le même ID utilisateur que celui sous lequel le processus d'agent s'exécute.
- Membres du groupe spécifié par la propriété d'agent **adminGroup**.

Pour plus d'informations, voir la propriété **adminGroup** dans «Le fichier MFT `agent.properties`», à la page 564.

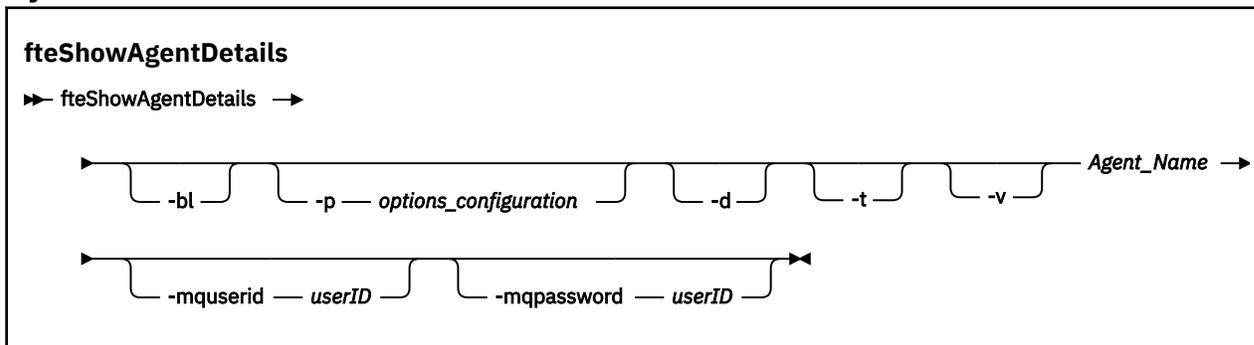
Pour IBM WebSphere MQ 7.5 ou version ultérieure, le statut du contrôleur de processus d'agent et du gestionnaire de files d'attente est disponible si vous exécutez la commande sur le même système que l'agent. Vous pouvez utiliser ces informations pour vous aider à identifier les problèmes. De plus, lorsque vous exécutez la commande sur le même système que l'agent, des informations de statut d'agent plus détaillées sont disponibles pour le cas où l'agent s'est arrêté de manière inattendue.

Pour obtenir la liste des valeurs de statut d'agent possibles et leur signification, voir «Valeurs de statut de l'agent MFT», à la page 642.

Pour obtenir la liste des valeurs de statut possibles pour le contrôleur de processus d'agent et leur signification, voir «Valeurs de statut du contrôleur de processus de l'agent MFT», à la page 645.

Pour obtenir la liste des valeurs de trace d'agent et des spécifications FFDC, ainsi que leur signification, voir «`fteSetAgentTraceLevel`», à la page 280

Syntaxe



Paramètre

-bl

Facultatif. Génère également le niveau de compilation du produit pour l'agent.

-p (*options_configuration*)

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour émettre la demande d'affichage des détails d'un agent. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

-d

Facultatif. Indique que les informations de diagnostic sont collectées pour *nom_agent*.

Les informations de diagnostic sont envoyées à la console et écrites dans un fichier appelé `diagnostics.<yyyyMMdd>.<HHmmss>.<ssss>.<number>.properties` dans le répertoire `MQ_DATA_PATH\mqft\logs\coordination_qmgr_name\agents\agent_name\logs`. Un maximum de cinq fichiers d'historique contenant des informations de diagnostic sur un agent seront créés. Si cinq fichiers d'historique ont été créés pour un agent lorsque la commande **fteShowAgentDetails** est exécutée avec le paramètre **-d** spécifié, le fichier d'historique le plus ancien est supprimé et remplacé par un nouveau fichier contenant les dernières informations de diagnostic sur l'agent.

Vous pouvez utiliser ce paramètre uniquement lorsque l'agent est en cours d'exécution et sur le système local.

-t

Facultatif. Spécifie le mode concis. Sous IBM MQ 9.0.2, la sortie inclut les informations **Status Age** par défaut. Si vous ne souhaitez pas voir ces informations, vous pouvez exécuter la commande avec le paramètre **-t**. Pour plus d'informations, voir [«Que faire si un agent est affiché comme étant à l'état UNKNOWN \(inconnu\)»](#), à la page 311.

-v

Facultatif. Indique le mode prolixe, qui génère une sortie supplémentaire pour l'agent. Il s'agit notamment du nom d'hôte, de la version du produit, du niveau de compilation du produit, du niveau de trace et de la spécification FFDC (First Failure Data Capture), ainsi que d'une liste des états de transfert pour chaque transfert source et cible en cours.

Les informations de transfert en cours sont obtenues à partir de la publication de statut de l'agent, décrite dans [«Format de message d'état de l'agent MFT»](#), à la page 757. Par conséquent, ces informations de transfert ne sont exactes que dans la limite de la valeur de la propriété `agentStatusPublishRateLimit`. Pour plus de détails sur cette propriété, voir [«Le fichier MFT agent.properties»](#), à la page 564.

Agent_Name

Obligatoire. Nom du Managed File Transfer Agent que vous souhaitez afficher.

-mquserid (userID)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de file d'attente de coordination.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de coordination. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans l'exemple suivant, en exécutant l'agent de liaisons, en exécutant la commande **fteShowAgentDetails** localement sur l'agent:

```
fteShowAgentDetails -v AGENT1
```

```
5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
```

Agent Information:

```
Name: AGENT1
Type: Standard
Description:
Operating System: Windows Server 2003
Time Zone: Greenwich Mean Time
Product Version: 7.5
Build Level: f000-20120312-0957
Trace Level: com.ibm.wmqfte.Agent=all
com.ibm.wmqfte.common=all
Trace FFDC: com.ibm.wmqfte.common:Any
com.ibm.wmqfte.Agent:1
```

Agent Controller Information:

```
Controller type: MQMFT Process Controller
Status: STARTED
Status Details: The agent process controller has
started the agent process.
Agent Restarts within Interval: 0
Total Agent Restart Count: 0
```

Agent Availability Information:

```
Status: READY
Status Details: The agent is running and is publishing
its status at regular intervals. The
last update was received within the
expected time period. The agent is
ready to process transfers, but none
are currently in progress.
```

Queue Manager Information:

```
Name: QM1
Transport: Bindings
Last Status Reported: AVAILABLE (Last Error MQRC: 0)
Status Details: The queue manager is available.
```

```
Maximum Number of Running Source Transfers: 25
Maximum Number of Queued Source Transfers: 1000
Source Transfer States:
No current transfers
```

Maximum Number of Running Destination Transfers: 25

Destination Transfer States:

```
TransferId State
414d51204d49414f5720202020202020202020822c5b4a648c0b20 progress
414d51204d49414f5720202020202020202020822c5b4a346c0b20 progress
```

Dans l'exemple suivant, QMGR1 est le gestionnaire de files d'attente de coordination utilisé comme entrée pour les options de configuration et les informations de diagnostic sont demandées avec le paramètre **-d**. La commande **fteShowAgentDetails** est émise sur un système IBM WebSphere MQ 7.5.0 Fix Pack 1 avec un agent local:

```
fteShowAgentDetails -p QMGR1 -d AGENT1
5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
Agent Information:
  Name: AGENT1
  Type: Standard
  Description:
  Operating System: Linux
  Time Zone: Greenwich Mean Time

Agent Controller Information:
  Controller type: MQMFT Process Controller
  Status: STARTED
  Status Details: The agent process controller has started
                  the agent process.
  Agent Restarts within Interval: 0
  Total Agent Restart Count: 0

Agent Availability Information:
  Status: ACTIVE
  Status Details: The agent is running and is publishing
                  its status at regular intervals. The last
                  update was received within the expected
                  time period. The agent is currently
                  processing one or more transfers.

Queue Manager Information:
  Name: QMGR1
  Transport: Client
  Host: host1.hursley.ibm.com
  Port: 2021
  Channel: SYSTEM.DEF.SVRCONN
  Last Status Reported: UNKNOWN
  Status Details: Information about the queue manager is
                  not available because the agent has a
                  client connection to the queue manager.

Agent Diagnostic Information:

Command Handler Diagnostics:
  Last Command Queue Read Time: 2012-07-30T15:23:10.705Z
  Pending Command Queue Size: 0
  Last Internal Command Type: Resync Request (from sender) -
414d5120514d43414e4445202020202079e20f5064230010
  Last Internal Command Time: 2012-07-30T14:17:10.506Z
  Last External Command Type: New Monitor Request
  Last External Command Time: 2012-07-30T14:10:57.751Z
  Diagnostic Properties File name: C:\Program Files (x86)\IBM\WebSphere
MQ\mqft\logs\MUNGEE\agents\MUNGEE\logs\di
agnostics.20121031.083420.0477.1.properti
es

Command Handler Worker Thread 0 Diagnostics:
  Status: Waiting

Command Handler Worker Thread 1 Diagnostics:
  Status: Waiting

Command Handler Worker Thread 2 Diagnostics:
  Status: Waiting

Command Handler Worker Thread 3 Diagnostics:
  Status: Waiting

Command Handler Worker Thread 4 Diagnostics:
  Status: Waiting

File Transfer Diagnostics:
  Source Transfers: 1
  Destination Transfers: 2

File Transfer 0 Diagnostics:
  Transfer Id: 414d5120514d43414e4445202020202079e20f5064230010
  Role: SOURCE
  State: ReSynchronisingTransfer
  Status: INACTIVE
```

```

Start Time:                Not started
Retry Count:               0
CheckPoint Index:         0
CheckPoint Position:      0

File Transfer 1 Diagnostics:
Transfer Id:               414d5120514d43414e44452020202020c8fbd54f144f0d20
Role:                     DESTINATION
State:                     RunningTransfer
CheckPoint Index:         0
CheckPoint Position:      0
Write Index:               0
Write Position:            0

File Transfer 2 Diagnostics:
Transfer Id:               414d5120514d43414e4445202020202079e20f5086020010
Role:                     DESTINATION
State:                     RunningTransfer
CheckPoint Index:         9
CheckPoint Position:      0
Write Index:               3
Write Position:            140923

Monitor 0 Diagnostics:
Name:                     MONITOR1
Status:                   STARTED
Resource Type:            directory
Resource:                 /tmp/monitor
Poll Interval:            1 minutes
Batch Size:               2
Condition:                Match
Pattern:                  * (wildcard)
Executing:                 false
Last Execute Start Time:  2012-04-04T16:19:01.852Z
Last Execute End Time:    2012-04-04T16:19:01.852Z
Last Execute Match Count: 0

Schedule 1 Diagnostics:
Id:                       1
Next Trigger Time:        2012-07-17T16:00+0100
Occurrences So Far:       14
Repeat Interval:          hours
Repeat Frequency:         5
Source Agent:             AGCANDE
Destination Agent:        AGCANDE
Source File:              /tmp/source/a.txt, ...
Destination File:         /tmp/dest/a.txt, ...

```

Dans l'exemple suivant, l'agent de liaisons a été arrêté et la commande **fteShowAgentDetails** a été émise à distance à partir de l'agent:

```

fteShowAgentDetails AGENT2
5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
Agent Information:
  Name:                    AGENT2
  Type:                    Standard
  Description:
  Operating System:        Linux
  Time Zone:               Greenwich Mean Time

Agent Controller Information:
  Controller type:         MQMFT Process Controller
  Status:                  UNKNOWN
  Status Details:         Information about the agent controller
                           is not available, either because the
                           agent is not running or the agent is
                           running on a different system.

  Agent Restarts within Interval: 0
  Total Agent Restart Count:      0

Agent Availability Information:
  Status:                  STOPPED
  Status Details:         The agent has been stopped. It was shut
                           down in a controlled manner.

Queue Manager Information:
  Name:                    QM2
  Transport:               Bindings
  Last Status Reported:    UNKNOWN
  Status Details:         Information about the queue manager is

```

```
not available, either because the agent
is not running or the agent is running
on a different system.
```

Dans l'exemple suivant, l'agent de liaison est en attente de redémarrage avec le gestionnaire de files d'attente d'agent arrêté. L'agent a déjà été redémarré une fois avant `Total Agent Restart Count: 1`, probablement en raison d'un redémarrage précédent du gestionnaire de files d'attente de l'agent:

Remarque : `Last Error MQRC` par rapport à `Last Status Reported` pour les informations sur le gestionnaire de files d'attente ; ces informations sont conservées même lorsque le gestionnaire de files d'attente devient disponible.

```
fteShowAgentDetails AGENT1
5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
Agent Information:
  Name: AGENT1
  Type: Standard
  Description:
  Operating System: Windows Server 2003
  Time Zone: Greenwich Mean Time

Agent Controller Information:
  Controller type: MQMFT Process Controller
  Status: WAITING
  Status Details: The agent process controller is waiting
                  for the queue manager to become
                  available before starting the agent.

  Agent Restarts within Interval: 0
  Total Agent Restart Count: 1

Agent Availability Information:
  Status: STOPPED
  Status Details: The agent has been stopped. It was shut
                  down in a controlled manner.

Queue Manager Information:
  Name: QM1
  Transport: Bindings
  Last Status Reported: UNAVAILABLE (Last Error MQRC: 2059)
  Status Details: The queue manager is unavailable. It
                  might be that the queue manager has not
                  been started or an incorrect queue
                  manager name has been configured. Look
                  up the MQ reason code reported against
                  the status to understand the problem.
```

Dans l'exemple suivant, l'agent en mode client vient de s'arrêter de manière inattendue et le contrôleur de processus de l'agent tente de récupérer la situation en la redémarrant après un délai spécifié par la valeur de la propriété d'agent `maxRestartDelay`. La valeur par défaut de la propriété d'agent `maxRestartDelay` est -1, ce qui provoque l'arrêt du contrôleur de processus de l'agent. Par conséquent, dans cet exemple, la valeur de la propriété `maxRestartDelay` doit avoir été définie sur une valeur supérieure à 0. `Current Agent Restart Count: 4` implique qu'il y a eu 4 redémarrages au cours de la période de propriété de l'agent `maxRestartInterval`. Si la propriété d'agent `maxRestartCount` est 4, après 4 redémarrages dans `maxRestartInterval`, le contrôleur de processus de l'agent attend `maxRestartDelay` secondes avant de redémarrer l'agent, ce qui est le cas ici. Le `Total Agent restart Count: 8` suggère que cela s'est produit auparavant. Cet exemple n'est pas typique et vous ne pouvez vous attendre à voir l'agent s'arrêter de manière inattendue que si l'agent est à court de mémoire ou qu'un exit utilisateur personnalisé a provoqué une sorte d'erreur d'exécution. Des détails complets sur la raison pour laquelle l'agent s'est arrêté de manière inattendue figurent dans le fichier `output0.log` de l'agent:

```
fteShowAgentDetails AGENT3
5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
Agent Information:
  Name: AGENT3
  Type: Standard
  Description:
  Operating System: Windows Server 2003
  Time Zone: Greenwich Mean Time

Agent Controller Information:
  Controller type: MQMFT Process Controller
```

```

Status: RECOVERING
Status Details: The agent process unexpectedly stopped
and the process controller will attempt
to restart it.

Current Agent Restart Count: 4
Total Agent Restart Count: 8

Agent Availability Information:
Status: ENDED UNEXPECTEDLY
Status Details: The agent has ended unexpectedly due to
an unrecoverable problem. The agent
will be automatically restarted.

Queue Manager Information:
Name: QM3
Transport: Client
Host: host3.hursley.ibm.com
Port: 3031
Channel: SYSTEM.DEF.SVRCONN

```

Dans l'exemple suivant, les résultats d'un agent de pont Connect:Direct sont affichés:

```

fteShowAgentDetails AG_CD1
5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
Agent Information:
Name: AG_CD1
Type: Connect:Direct bridge
Description:
Connect:Direct Node Name: CDNODE
Connect:Direct Node Host: localhost:1363
Operating System: Windows Server 2003
Time Zone: Greenwich Mean Time

Agent Controller Information:
Controller type: MQMFT Process Controller
Status: UNKNOWN
Status Details: Information about the agent controller
is not available, either because the
agent is not running or the agent is
running on a different system.

Agent Restarts within Interval: 0
Total Agent Restart Count: 0

Agent Availability Information:
Status: STOPPED
Status Details: The agent has been stopped. It was shut
down in a controlled manner.

Queue Manager Information:
Name: QM_JUPITER
Transport: Bindings
Last Status Reported: UNKNOWN
Status Details: Information about the queue manager is
not available, either because the agent
is not running or the agent is running
on a different system.

```

 Dans l'exemple suivant, un agent s'exécutant sur z/OS est enregistré auprès du gestionnaire ARM (Automatic Restart Manager):

```

fteShowAgentDetails AGENTZ
5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
Agent Information:
Name: AGENTZ
Type: Standard
Description:
Operating System: z/OS
Time Zone: Greenwich Mean Time

Agent Controller Information:
Controller Type: z/OS Automatic Restart Manager (ARM)
Agent registered with ARM: Yes (ELEMENTYPE: SYSBFGAG, ELEMENT: AGENTZ)
Agent Restarted: No

Agent Availability Information:
Status: READY
Status Details: The agent is running and is publishing
its status at regular intervals. The last

```

```
update was received within the expected
time period. The agent is ready to
process transfers, but none are currently
in progress.
```

```
Queue Manager Information:
Name: ZQM
Transport: Bindings
Last Status Reported: AVAILABLE
Status Details: The queue manager is available.
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Référence associée

[«Agents fteList: répertorie les agents MFT pour un gestionnaire de files d'attente de coordination», à la page 484](#)

La commande **fteListAgents** permet de répertorier tous les agents Managed File Transfer enregistrés auprès d'un gestionnaire de files d'attente de coordination particulier.

[«Valeurs de statut de l'agent MFT», à la page 642](#)

Les commandes **fteListAgents** et **fteShowAgentDetails** produisent des informations sur le statut de l'agent. Il existe plusieurs valeurs possibles pour ce statut.

[«Valeurs de statut du contrôleur de processus de l'agent MFT», à la page 645](#)

La commande **fteShowAgentDetails** génère des informations sur le statut du contrôleur de processus de l'agent. Il existe plusieurs valeurs possibles pour ce statut.

fteShowLoggerDetails

La commande **fteShowLoggerDetails** permet d'afficher les détails d'un consignateur Managed File Transfer particulier.

Objet

Vous devez exécuter la commande **fteShowLoggerDetails** sur le même système que le consignateur. Il affiche le statut du contrôleur de processus du consignateur et du gestionnaire de files d'attente du consignateur, que vous pouvez utiliser pour faciliter l'identification des incidents. La commande **fteShowLoggerDetails** répertorie les détails suivants pour un consignateur Managed File Transfer particulier:

- Statut du contrôleur de consignateur.
- Le consignateur redémarre dans l'intervalle
- Nombre total de redémarrages du consignateur
- Statut de disponibilité du consignateur
- Nom du gestionnaire de files d'attente du consignateur
- Type de transport du gestionnaire de files d'attente du consignateur
- Dernier statut du gestionnaire de files d'attente du consignateur signalé (s'applique uniquement au mode de transport de liaison)

Spécifiez le paramètre facultatif **-p** pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de votre ensemble par défaut. Pour plus d'informations, voir [«Options de configuration de MFT sur Multiplatforms», à la page 58](#).

Pour obtenir la liste des valeurs de statut de consignateur possibles et leur signification, voir [«Valeurs de statut du consignateur MFT», à la page 645](#).

Pour obtenir la liste des valeurs de statut possibles pour le contrôleur de processus du consignateur et leur signification, voir «Valeurs de statut du contrôleur de processus du consignateur MFT», à la page 646.

V 9.0.4 Depuis la IBM MQ 9.0.4, la sortie de la commande affiche les informations de connexion que le consignateur utilise pour se connecter au gestionnaire de files d'attente.

V 9.0.4 Si le consignateur est connecté en mode clients, la sortie pour:

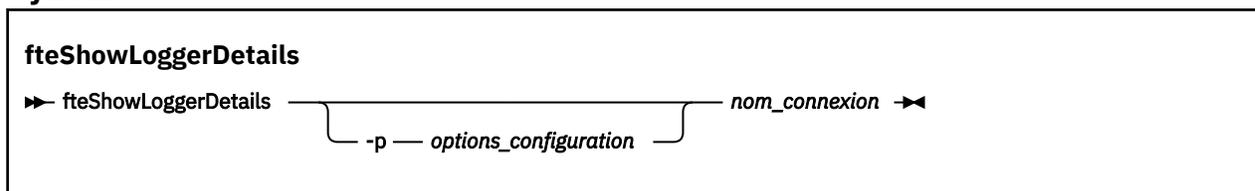
Dernier statut signalé

S'affiche sous la forme UNKNOWN

Détails sur le statut

S'affiche sous la forme Information about the queue manager is not available because the logger has a client connection to queue manager.

Syntaxe



Paramètre

-p *options_configuration*

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour émettre la demande d'affichage des détails d'un consignateur. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

nom_consignateur

Obligatoire. Nom du consignateur Managed File Transfer que vous souhaitez afficher.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, un consignateur démarré, émettant la commande **fteShowLoggerDetails** localement sur le consignateur:

```
fteShowLoggerDetails LOGGER1
```

```
5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
Logger Controller Information:
  Status:                STARTED
  Status Details:        The logger process controller has
                        started the logger process.
  Logger Restarts within Interval: 0
  Total Logger Restart Count:    0

Queue Manager Information:
  Name:                  QM_gbthink
  Transport:             Bindings
  Last Status Reported:  AVAILABLE
  Status Details:        The queue manager is available.
```

Dans cet exemple, un consignateur attend en raison d'un gestionnaire de files d'attente non disponible, en émettant la commande **fteShowLoggerDetails** localement sur le consignateur:

```
fteShowLoggerDetails LOGGER2
```

```
5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
Logger Controller Information:
  Status: WAITING
  Status Details: The logger process controller is
                  waiting for the queue manager to
                  become available before starting the
                  logger.
  Logger Restarts within Interval: 0
  Total Logger Restart Count: 0

Logger Availability Information:
  Status: STOPPED
  Status Details: The logger has been stopped. It was
                  shut down in a controlled manner.

Queue Manager Information:
  Name: QM_gbthink
  Transport: Bindings
  Last Status Reported: UNAVAILABLE (Last Error MQRC: 2059)
  Status Details: The queue manager is unavailable. It
                  might be that the queue manager has
                  not been started or an incorrect
                  queue manager name has been
                  configured. Look up the MQ reason code
                  reported against the status to
                  understand the problem.
```

Dans cet exemple sous z/OS, un consignateur en cours d'exécution (non enregistré auprès d'ARM):

```
fteShowLoggerDetails loggerv8
```

```
5655-MFT, 5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
Logger Controller Information:
  Controller Type: z/OS Automatic Restart Manager (ARM)
  Registered with ARM: No
  Restarted: n/a

Queue Manager Information:
  Name: FT8E
  Transport: Bindings
  Last Status Reported: AVAILABLE
  Status Details: The queue manager is available.
```

Dans cet exemple sous z/OS, un consignateur qui n'est pas en cours d'exécution ou qui s'exécute sur un autre système:

```
fteShowLoggerDetails loggerv8
```

```
5655-MFT, 5724-H72 Copyright IBM Corp. 2008, 2023. ALL RIGHTS RESERVED
Logger Controller Information:
  Controller Type: UNKNOWN

Queue Manager Information:
  Name: FT8E
  Transport: Bindings
  Last Status Reported: UNKNOWN
  Status Details: Information about the queue manager is
                  not available, either because the
                  logger is not running, or the logger
                  is running on a different system.
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Référence associée

«Valeurs de statut du consignateur MFT», à la page 645

Les commandes **fteShowLoggerDetails** produisent des informations sur le statut du consignateur. Il existe plusieurs valeurs possibles pour ce statut.

«Valeurs de statut du contrôleur de processus du consignateur MFT», à la page 646

La commande **fteShowLoggerDetails** génère des informations sur le statut du contrôleur de processus du consignateur. Il existe plusieurs valeurs possibles pour ce statut.

fteStartAgent

La commande **fteStartAgent** démarre un agent Managed File Transfer à partir de la ligne de commande.

Objet

Utilisez la commande **fteStartAgent** pour démarrer un agent Managed File Transfer . Vous devez démarrer un agent avant de pouvoir l'utiliser pour effectuer des transferts de fichiers. La commande **fteStartAgent** démarre un agent sur le système sur lequel vous émettez la commande: vous ne pouvez pas démarrer un agent sur un système distant.

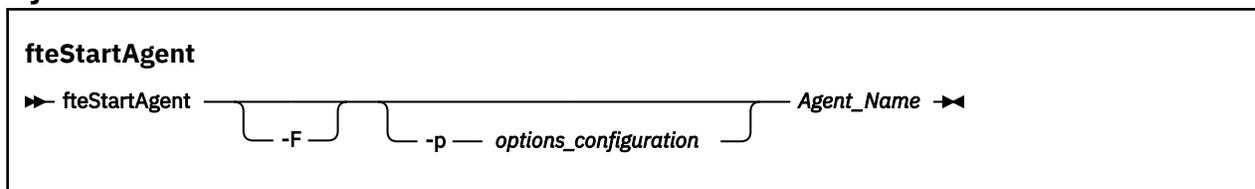
Pour IBM WebSphere MQ 7.5 ou version ultérieure, le contrôleur de processus de l'agent gère le démarrage de l'agent. Toutefois, le contrôleur de processus de l'agent peut attendre un certain temps, par exemple lorsque le taux d'échecs de l'agent est élevé, avant de tenter de redémarrer l'agent. En tant qu'administrateur IBM MQ , vous pouvez utiliser la commande **fteStartAgent** pour remplacer cette attente et lancer un démarrage de l'agent. Si le contrôleur de processus d'agent attendait que le gestionnaire de files d'attente soit disponible, cette commande lance également le contrôleur de processus d'agent qui tente de se reconnecter au gestionnaire de files d'attente.

Windows Si vous avez configuré l'agent pour qu'il s'exécute en tant que service Windows à l'aide de la commande [fteCreateAgent](#) ou [fteModifyAgent](#) , l'exécution de la commande **fteStartAgent** démarre le service Windows .

Cette commande renvoie une erreur si l'agent ne démarre pas ou qu'il est déjà démarré. L'agent communique avec son gestionnaire de files d'attente en fonction des valeurs définies dans le fichier `agent.properties` .

Spécifiez le paramètre facultatif **-p** pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de celui défini par défaut. Pour plus d'informations, voir «Le fichier MFT `agent.properties`», à la page 564.

Syntaxe



Paramètre

-F

Facultatif. Ce paramètre exécute le démon d'agent en tant que processus d'avant-plan. Par défaut, le démon d'agent s'exécute en arrière-plan.

Si vous travaillez sous Windows et que vous avez configuré l'agent pour qu'il s'exécute en tant que service Windows à l'aide des commandes **fteCreateAgent** ou **fteModifyAgent**, le paramètre **-F** remplace cette configuration.

-p options_configuration

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour émettre la demande de démarrage d'un agent. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

Agent_Name

Obligatoire. Nom de l'agent Managed File Transfer que vous souhaitez démarrer.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, AGENT2 est démarré et s'exécute en avant-plan.

```
fteStartAgent -F AGENT2
```

Dans l'exemple suivant (pour les systèmes UNIX et Linux), AGENT2 est démarré avec un gestionnaire de files d'attente de coordination autre que celui par défaut, QM_SATURN:

```
./fteStartAgent -p QM_SATURN AGENT2
```

Vous pouvez également exécuter la commande en spécifiant le chemin d'accès à **fteStartAgent** comme suit:

```
path/fteStartAgent agentname
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Réponses

Dans certains cas, des messages d'erreur peuvent s'afficher après l'exécution de la commande **fteStartAgent** :

- Si vous exécutez la commande **fteStartAgent** et que le message d'erreur suivant s'affiche, votre environnement contient probablement des chemins d'accès à la bibliothèque supplémentaires qui entrent en conflit avec Managed File Transfer :

```
BFGCL0001E: Une erreur interne s'est produite. Exception : 'CC=2;RC=2495;AMQ8568:  
La bibliothèque JNI native « mqjnd » est introuvable. [3=mqjnd]
```

Si la variable d'environnement LD_LIBRARY_PATH ou LIBPATH est définie pour référencer une version 64 bits de la bibliothèque avant la version 32 bits lorsque l'agent est exécuté avec une version 32 bits de Java (comme c'est actuellement le cas de la plupart des plateformes), cette erreur se produit.

Pour résoudre ce problème, définissez la propriété `javaLibraryPath` de l'agent Managed File Transfer afin qu'elle référence l'emplacement correct de la bibliothèque. Par exemple, pour `mqjbnd` sur AIX, définissez la valeur sur : `/usr/mqm/java/lib`. Pour `mqjbnd` sous Linux, définissez sur : `/opt/mqm/java/lib`

Tâches associées

«[Démarrage d'un agent MFT en tant que service Windows](#)», à la page 159

Vous pouvez démarrer un agent en tant que service Windows de sorte que lorsque vous vous déconnectez de Windows, votre agent continue de s'exécuter et puisse recevoir des transferts de fichiers.

«[Liste des agents MFT](#)», à la page 217

Vous pouvez répertorier les agents Managed File Transfer enregistrés auprès d'un gestionnaire de files d'attente particulier à l'aide de la ligne de commande ou de l' IBM MQ Explorer.

«[Arrêt d'un agent MFT](#)», à la page 218

Vous pouvez arrêter un agent Managed File Transfer à partir de la ligne de commande. Lorsque vous arrêtez un agent, vous mettez l'agent au repos et vous l'autorisez à terminer son transfert de fichiers en cours avant de l'arrêter. Vous pouvez également spécifier le paramètre **-i** sur la ligne de commande pour arrêter immédiatement un agent. Une fois l'agent arrêté, vous ne pouvez pas l'utiliser pour transférer des fichiers tant que vous ne l'avez pas redémarré.

Référence associée

«[Démarrage d'un agent MFT sous z/OS](#)», à la page 162

Sous z/OS, en plus d'exécuter la commande **fteStartAgent** à partir d'une session UNIX System Services, vous pouvez démarrer un agent en tant que tâche démarrée à partir de JCL sans avoir besoin d'une session interactive.

fteStartLogger

La commande **fteStartLogger** démarre une application de journalisation Managed File Transfer .

Objet

Utilisez la commande **fteStartLogger** pour démarrer un consignateur. Le consignateur peut être un fichier ou une application de base de données qui s'exécute sur le même système que le gestionnaire de file d'attente de coordination. Pour plus d'informations, voir «[Configuration d'un consignateur MFT](#)», à la page 107. Pour IBM WebSphere MQ 7.5 ou version ultérieure, le contrôleur de processus du consignateur gère le démarrage du consignateur. Toutefois, le contrôleur de processus du consignateur peut attendre un certain temps, par exemple lorsqu'il y a eu un taux élevé d'échecs du consignateur, avant de tenter de redémarrer le consignateur. En tant qu'administrateur IBM MQ , vous pouvez utiliser la commande **fteStartLogger** pour remplacer cette attente et lancer un démarrage du consignateur. Si le contrôleur de processus du consignateur attendait que le gestionnaire de files d'attente soit disponible, cette commande lance également le contrôleur de processus du consignateur en tentant de se reconnecter au gestionnaire de files d'attente.

Si vous avez configuré un consignateur pour qu'il s'exécute en tant que service Windows à l'aide de la commande `fteModifyLogger` , l'exécution de la commande **fteStartLogger** démarre le service Windows .

Cette commande renvoie une erreur si le consignateur ne démarre pas ou est déjà démarré. Le consignateur communique avec son gestionnaire de files d'attente en fonction des valeurs définies dans le fichier `logger.properties` .

Spécifiez le paramètre **-p** pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de la valeur par défaut. Pour plus d'informations sur les propriétés du consignateur, voir «[Propriétés de configuration du consignateur MFT](#)», à la page 120

Syntaxe

fteStartLogger

► fteStartLogger — *options_configuration* — -F — logger_name ►

Paramètres

nom_consignateur

Obligatoire. Nom du consigneateur Managed File Transfer que vous souhaitez démarrer.

-p options_configuration

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour émettre la demande de démarrage d'un consigneateur. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. **fteStartLogger** utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

-F

Facultatif. Exécute le consigneateur en tant que processus d'avant-plan (plutôt qu'en tant que processus d'arrière-plan par défaut). Si vous avez configuré le consigneateur pour qu'il s'exécute en tant que service Windows à l'aide de la commande **fteModifyLogger**, le paramètre **-F** remplace cette configuration.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, un consigneateur nommé logger1a été créé précédemment. Cette commande montre comment le consigneateur peut être démarré en tant que processus d'avant-plan:

```
fteStartLogger -F logger1
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Concepts associés

[«Configuration d'un consigneateur MFT», à la page 107](#)

Référence associée

[«fteModifyLogger \(exécutez un consigneateur MFT en tant que service Windows\)», à la page 504](#)

La commande **fteModifyLogger** permet de modifier un consigneateur Managed File Transfer afin qu'il puisse être exécuté en tant que service Windows. Vous pouvez utiliser cette commande uniquement sur les plateformes Windows, vous devez être exécuté par un utilisateur qui est un administrateur IBM MQ et un membre du groupe mqm, et vous devez d'abord arrêter le consigneateur à l'aide de la commande **fteStopLogger**.

[«fteStopLogger», à la page 546](#)

La commande **fteStopLogger** arrête un consigneateur Managed File Transfer.

[«Traitement et rejet des erreurs du consigneateur MFT», à la page 339](#)

Le consignateur Managed File Transfer identifie deux types d'erreur: les erreurs par message et les erreurs générales.

fteStopAgent

Utilisez la commande **fteStopAgent** pour arrêter un agent Managed File Transfer de manière contrôlée ou pour arrêter un agent immédiatement si nécessaire à l'aide du paramètre **-i**.

Objet

Lorsque vous arrêtez un agent à l'aide de la commande **fteStopAgent**, vous pouvez soit autoriser l'agent à terminer son transfert de fichier en cours avant de l'arrêter, soit l'arrêter immédiatement même si l'agent est en train de transférer un fichier. Une fois l'agent arrêté, vous ne pouvez pas l'utiliser pour transférer des fichiers tant que vous n'avez pas redémarré l'agent.

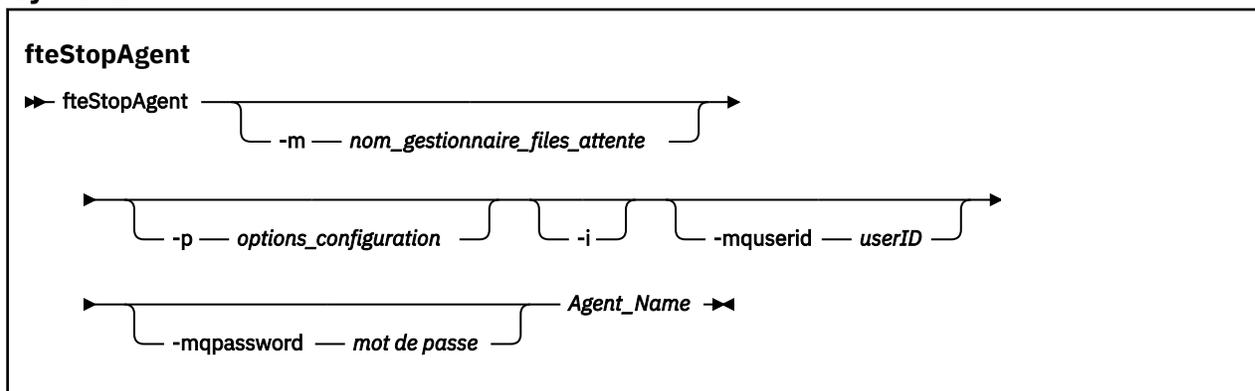
Si l'agent que vous souhaitez arrêter est connecté au réseau IBM MQ, vous pouvez exécuter la commande **fteStopAgent** à partir de n'importe quel système pouvant se connecter au réseau IBM MQ et être acheminé vers le gestionnaire de files d'attente de l'agent. Pour que la commande puisse être exécutée, vous devez avoir installé et configuré un composant Managed File Transfer (Service ou Agent) sur ce système pour communiquer avec le réseau IBM MQ. Si aucun détail de connectivité n'est disponible, une connexion en mode liaisons est établie avec le gestionnaire de files d'attente par défaut sur le système local. Si `command.properties` n'existe pas, une erreur est générée.

Si l'agent que vous souhaitez arrêter n'est pas connecté au réseau IBM MQ, par exemple si le réseau IBM MQ n'est pas actuellement disponible, vous pouvez uniquement exécuter la commande **fteStopAgent** à partir du système sur lequel l'agent s'exécute. Pour arrêter un agent qui n'est pas connecté au réseau IBM MQ, vous devez exécuter la commande **fteStopAgent** à partir du même utilisateur que l'agent. Sinon, si l'agent s'exécute sur un système Windows, vous pouvez exécuter la commande en tant qu'administrateur.

Spécifiez le paramètre facultatif **-p** pour cette commande uniquement si vous souhaitez utiliser un ensemble d'options de configuration différent de votre ensemble par défaut. Pour plus d'informations, voir «Le fichier MFT agent.properties», à la page 564.

Si votre agent s'exécute en tant que service Windows, l'exécution de la commande **fteStopAgent** arrête le service Windows. Pour plus d'informations, voir la section «Démarrage d'un agent MFT en tant que service Windows», à la page 159.

Syntaxe



Paramètres

-m (agent_qmgr_name)

Facultatif. Nom du gestionnaire de files d'attente auquel l'agent que vous souhaitez arrêter est connecté.

Si l'agent se trouve sur un système distant ou si l'agent se trouve sur le système local mais que vous n'êtes pas l'utilisateur qui l'a démarré, vous devez utiliser le paramètre **-m** et disposer des droits appropriés. Pour plus d'informations sur les droits d'accès, voir [«Restriction des droits de groupe pour les ressources spécifiques à MFT»](#), à la page 362.

-p (options_configuration)

Facultatif. Ce paramètre détermine l'ensemble des options de configuration utilisées pour émettre la demande d'arrêt d'un agent. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme entrée pour ce paramètre. La commande utilise ensuite l'ensemble des fichiers de propriétés associés à ce gestionnaire de files d'attente de coordination autre que celui par défaut.

Si vous ne spécifiez pas ce paramètre, l'ensemble des options de configuration basées sur le gestionnaire de file d'attente de coordination par défaut est utilisé.

-i

Facultatif. Arrête immédiatement l'agent. L'agent n'effectue aucun transfert en cours.

Si vous ne spécifiez pas le paramètre **-i**, l'agent effectue tous les transferts en cours mais ne démarre aucun nouveau transfert.

-mquserid (userID)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de files d'attente de commandes.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de commandes. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

Agent_Name

Obligatoire. Nom de l'agent Managed File Transfer que vous souhaitez arrêter.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, l'agent AGENT2 sur le gestionnaire de files d'attente QM_JUPITER est arrêté. Le paramètre **-m** est utilisé car le gestionnaire de files d'attente auquel AGENT2 est connecté est différent du gestionnaire de files d'attente spécifié par l'ensemble d'options de configuration.

```
fteStopAgent -m QM_JUPITER AGENT2
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

Tâches associées

[«Arrêt d'un agent MFT»](#), à la page 218

Vous pouvez arrêter un agent Managed File Transfer à partir de la ligne de commande. Lorsque vous arrêtez un agent, vous mettez l'agent au repos et vous l'autorisez à terminer son transfert de fichiers en cours avant de l'arrêter. Vous pouvez également spécifier le paramètre **-i** sur la ligne de commande pour arrêter immédiatement un agent. Une fois l'agent arrêté, vous ne pouvez pas l'utiliser pour transférer des fichiers tant que vous ne l'avez pas redémarré.

Référence associée

«fteStartAgent», à la page 540

La commande **fteStartAgent** démarre un agent Managed File Transfer à partir de la ligne de commande.

«Arrêt d'un agent MFT sous z/OS», à la page 218

Si vous exécutez un Managed File Transfer Agent sous z/OS en tant que tâche démarrée à partir de JCL, l'agent accepte les commandes d'opérateur z/OS **MODIFY** et **STOP**, en plus de la commande **fteStopAgent**.

fteStopLogger

La commande **fteStopLogger** arrête un consigneur Managed File Transfer .

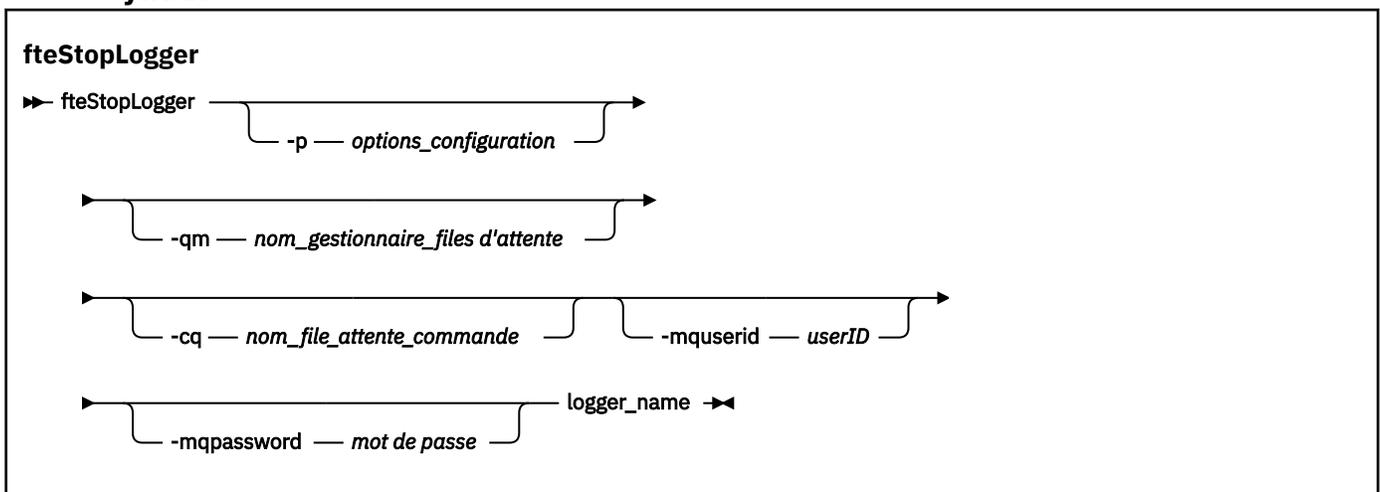
Objet

Utilisez la commande **fteStopLogger** pour arrêter un consigneur. Le consigneur peut être soit un consigneur de fichiers, qui enregistre un historique de l'activité de transfert de fichiers gérés dans un fichier, soit un consigneur de base de données qui enregistre l'historique dans une base de données.

Remarques supplémentaires sur l'arrêt d'un processus de consigneur autonome

Si votre consigneur s'exécute en tant que service Windows , l'exécution de la commande **fteStopLogger** arrête le service Windows .

Syntaxe



Paramètres

-p (*options_configuration*)

Facultatif. Détermine l'ensemble des options de configuration utilisées pour arrêter le consigneur. Utilisez le nom d'un ensemble d'options de configuration comme valeur pour le paramètre **-p**. Par convention, cette valeur correspond au nom d'un gestionnaire de files d'attente de coordination. Si vous ne spécifiez pas ce paramètre, l'ensemble d'options de configuration par défaut est utilisé.

-qm (*nom_gestionnaire_files d'attente*)

Facultatif. Par défaut, la file d'attente de commandes du consigneur est supposée se trouver dans le gestionnaire de file d'attente de coordination spécifié par le paramètre **-p** (ou sa valeur par défaut). Si vous souhaitez envoyer des commandes de consigneur à une file d'attente de commandes située ailleurs, utilisez le paramètre **-qm** pour spécifier une autre destination. Dans tous les cas, cette commande se connecte au gestionnaire de files d'attente de commandes indiqué par le paramètre **-p**, quelle que soit la destination finale du message.

-cq (nom_file_attente_commande)

Facultatif. Indique la file d'attente de commandes à laquelle envoyer le message d'arrêt. Dans la plupart des cas, les consignateurs utilisent le nom de file d'attente par défaut, ce qui signifie que ce paramètre n'est pas nécessaire.

-mquserid (userID)

Facultatif. Indique l'ID utilisateur pour l'authentification auprès du gestionnaire de files d'attente de commandes.

-mqpassword (mot de passe)

Facultatif. Indique le mot de passe à authentifier dans le gestionnaire de files d'attente de commandes. Vous devez également spécifier le paramètre **-mquserid**. Si vous spécifiez **-mquserid**, mais ne spécifiez pas **-mqpassword**, vous serez invité à fournir le mot de passe associé. Le mot de passe n'est pas affiché.

nom_consignateur

Obligatoire. Nom du consignateur Managed File Transfer que vous souhaitez arrêter.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, un consignateur a été créé précédemment sous le nom `logger1` et est en cours d'exécution. Cette commande montre comment le consignateur peut être arrêté:

```
fteStopLogger logger1
```

Codes retour**0**

La commande a abouti.

1

La commande a échoué.

Concepts associés

[«Configuration d'un consignateur MFT», à la page 107](#)

Référence associée

[«fteModifyLogger \(exécutez un consignateur MFT en tant que service Windows\)», à la page 504](#)

La commande **fteModifyLogger** permet de modifier un consignateur Managed File Transfer afin qu'il puisse être exécuté en tant que service Windows. Vous pouvez utiliser cette commande uniquement sur les plateformes Windows, vous devez être exécuté par un utilisateur qui est un administrateur IBM MQ et un membre du groupe `mqm`, et vous devez d'abord arrêter le consignateur à l'aide de la commande **fteStopLogger**.

[«fteStartLogger», à la page 542](#)

La commande **fteStartLogger** démarre une application de journalisation Managed File Transfer.

Référence de configuration MFT

Utilisation des variables d'environnement dans les propriétés MFT

Depuis la IBM WebSphere MQ 7.5, il est possible d'utiliser des variables d'environnement dans les propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction des changements d'environnement. Par exemple, l'utilisateur qui exécute le processus.

Les propriétés suivantes acceptent les emplacements de fichier ou de répertoire et peuvent donc contenir des variables d'environnement:

- agentSslKeyStore
- agentSslKeyStoreCredentialsFile
- agentSslTrustStore
- agentSslTrustStoreCredentialsFile
- cdNodeKeystoreCredentialsFile
- cdNodeTruststoreCredentialsFile
- cdTmpDir
- cdNodeKeystore
- cdNodeTruststore
- commandPath
- connectionSslKeyStore
- connectionSslKeyStoreCredentialsFile
- connectionSslTrustStore
- connectionSslTrustStoreCredentialsFile
- coordinationSslKeyStore
- coordinationSslKeyStoreCredentialsFile
- coordinationSslTrustStore
- coordinationSslTrustStoreCredentialsFile
- exitClassPath
- exitNativeLibraryPath
- javaCoreTriggerFile
- sandboxRoot
- transferRoot
- wmqfte.database.credentials.file

Exemple

Dans cet exemple sur un système Windows , un utilisateur `fteuser` utilisant une variable d'environnement `USERPROFILE`:

```
wmqfte.database.credentials.file=%USERPROFILE%\logger\mqmftcredentials.xml
```

Résout le chemin d'accès au fichier suivant:

```
C:\Users\fteuser\logger\mqmftcredentials.xml
```

Dans cet exemple sur un système UNIX , un utilisateur `fteuser` utilisant une variable d'environnement `HOME`:

```
transferRoot=$HOME/fte/
```

Résout le chemin d'accès au fichier suivant:

```
/home/fteuser/fte/
```

Le fichier MFT installation.properties

Le fichier `installation.properties` spécifie 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 de l'ensemble d'options de configuration correspond au nom du gestionnaire de files d'attente de coordination associé.

Ce fichier est créé par le programme d'installation et peut être modifié à l'aide de la commande **fteChangeDefaultConfigurationOptions**.

Le fichier `installation.properties` se trouve dans votre répertoire `MQ_DATA_PATH`.

Par exemple, sous Windows, l'emplacement de fichier par défaut est

`MQ_DATA_PATH\mqft\installations\installation_name` et sur les systèmes UNIX et Linux, l'emplacement de fichier par défaut est `/var/mqm/mqft/installations/installation_name`.

V 9.0.1 Pour l'Redistributable Managed File Transfer Agent, le chemin de données est défini lorsque vous exécutez la commande **fteCreateEnvironment**. Si vous exécutez la commande et indiquez l'emplacement de votre choix à l'aide du paramètre **-d**, le chemin de données est défini pour cet emplacement. Si vous ne spécifiez pas l'emplacement à l'aide de la commande **fteCreateEnvironment**, un répertoire `mftdata` est créé sous le répertoire racine de l'extraction de l'Redistributable Managed File Transfer Agent. Le fichier `installation.properties` pour Redistributable Managed File Transfer Agent se trouve dans le répertoire `MQ_DATA_PATH\mqft\installations\MFTZipInstall`.

Le fichier `installation.properties` contient les valeurs suivantes :

Tableau 39. Propriétés de base

Nom de la propriété	Description	Valeur par défaut
<p>commandMessagePriority</p>	<p>Définit la priorité des messages internes et des messages de commande des commandes fteStopAgent, fteCancelTransfer et ftePingAgent.</p> <p>Si, par exemple, vous soumettez un grand nombre de demandes de transfert pour transférer plein de petits fichiers par succession rapide, les nouvelles demandes de transfert peuvent être placées en file d'attente, dans la file d'attente des commandes de l'agent source. Les messages externes et internes possédant la priorité de message IBM MQ par défaut, les messages internes sont bloqués par les nouvelles demandes de transfert. Cela peut entraîner un dépassement du délai de négociation des transferts et un passage des transferts en récupération.</p> <p>Vous pouvez également utiliser la propriété <code>commandMessagePriority</code> pour définir la priorité des messages d'accusé de réception internes et d'accusé de réception attendu.</p> <p>Pour définir les priorités des messages Managed File Transfer internes au dessus des nouvelles demandes de transfert, définissez cette propriété sur une valeur comprise entre 1 (la plus faible) et 9 (la plus élevée).</p> <p>La valeur par défaut de la propriété <code>commandMessagePriority</code> est 8. Cela signifie que si l'attribut IBM MQ DEFPRTY (priorité par défaut) dans une file d'attente de commandes d'agent est inférieur ou égal à 7, les messages de négociation interne sont classés par priorité avant les nouvelles demandes de transfert. Si la valeur de l'attribut DEFPRTY est de 8 ou 9, pour conserver l'efficacité de la propriété <code>commandMessagePriority</code>, vous devez modifier DEFPRTY ou la propriété <code>commandMessagePriority</code>.</p>	<p>Pour IBM MQ 9.0.0.0, la valeur par défaut est 8.</p> <p>Pour les versions antérieures et avant le correctif APAR IT06213, la valeur par défaut est la constante <code>MQPRI_PRIORITY_AS_Q_DEF</code>, dont la valeur est -1.</p>

Tableau 39. Propriétés de base (suite)

Nom de la propriété	Description	Valeur par défaut
defaultProperties	Nom de l'ensemble par défaut d'options de configuration. Cette valeur correspond au nom d'un répertoire qui se trouve dans le répertoire de configuration, qui contient les répertoires et fichiers de propriétés qui spécifient les informations de configuration.	Aucune valeur par défaut
enableFunctionalFixPack	<p>Niveau de fonction du groupe de correctifs à activer. Par défaut, les nouvelles fonctions incluses dans un groupe de correctifs ne sont pas activées. Définissez cette propriété sur un identificateur de version pour activer les nouvelles fonctions disponibles avec cette version.</p> <p>Vous pouvez spécifier l'identificateur de version avec ou sans point (.). Par exemple, pour utiliser la fonction disponible avec IBM MQ 8.0.0 Fix Pack 2, définissez cette propriété sur 8002 ou 8.0.0.2.</p>	Aucune valeur par défaut

Tableau 39. Propriétés de base (suite)

Nom de la propriété	Description	Valeur par défaut
<p>     messagePublicationFormat </p>	<p>Permet de spécifier le format de publication des messages utilisé par les agents MFT pour leurs messages XML de statut. Cette propriété accepte les valeurs suivantes :</p> <p>messagePublicationFormat=mixed Les messages sont publiés sans FORMAT MQMD (MQFMT_NONE), excepté pour les messages publiés sous l'arborescence de rubriques /LOG, qui sont publiés au format MQFMT_STRING de MQMD.</p> <p>messagePublicationFormat=MQFMT_NONE Les messages sont publiés sans FORMAT MQMD.</p> <p>messagePublicationFormat=MQFMT_STRING Les messages sont publiés dans un format de chaîne.</p> <p> Il existe une erreur de codage dans la IBM MQ 9.0.2. Dans cette version, lorsque vous spécifiez messagePublicationFormat=mixed, cela revient à spécifier messagePublicationFormat=MQFMT_NONE.  L'erreur est corrigée dans la IBM MQ 9.0.3.</p> <p>Avant IBM MQ 8.0, les agents MFT publiait des messages de statut XML dans SYSTEM.FTE dans un format de chaîne (MQFMT_STRING). Si possible, les applications qui utilisaient précédemment la IBM WebSphere MQ 7.5 doivent être mises à jour pour traiter les messages au format de la IBM MQ 8.0 (ou ultérieure). S'il n'est pas possible de modifier une application, définissez la propriété messagePublicationFormat sur MQFMT_STRING pour rétablir le comportement de la IBM WebSphere MQ 7.5.</p>	<p>messagePublicationFormat=mixed</p>

Tableau 39. Propriétés de base (suite)

Nom de la propriété	Description	Valeur par défaut
 Spécifiques à z/OS :		
  productID	Type de produit sur lequel l'utilisation de MFT doit être enregistrée : <ul style="list-style-type: none"> • Produit Managed File Transfer autonome. (MFT correspond à productID). • Composant d'un produit IBM MQ Advanced. (ADVANCED correspond à productID). • Composant d'un produit IBM MQ Advanced for z/OS, Value Unit Edition. (ADVANCEDVUE correspond à productID). Pour plus d'informations sur l'enregistrement de l'utilisation du produit, voir Génération de rapports sur les informations produit .  Cette propriété est ignorée sous Multiplateformes .	MFT

Le texte suivant représente un exemple de contenu d'un fichier `installation.properties`.

```
defaultProperties=ERIS
```

ERIS représente le nom d'un répertoire qui se trouve dans le même répertoire que le fichier `installation.properties`. Le répertoire ERIS contient des répertoires et des fichiers de propriétés qui décrivent un ensemble d'options de configuration.

Concepts associés

«Options de configuration de MFT sur Multiplatforms», à la page 58

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

«`fteChangeDefaultConfigurationOptions`», à la page 393

Utilisez la commande **`fteChangeDefaultConfigurationOptions`** pour modifier les options de configuration par défaut que Managed File Transfer doit utiliser. La valeur des options de configuration définit le groupe de fichiers de propriétés utilisé par Managed File Transfer .

Le fichier MFT `coordination.properties`

Le fichier `coordination.properties` spécifie les détails de connexion au gestionnaire de files d'attente de coordination. Etant donné que plusieurs installations Managed File Transfer peuvent partager le même gestionnaire de files d'attente de coordination, vous pouvez utiliser un lien symbolique vers un fichier `coordination.properties` commun sur une unité partagée.

Le fichier `coordination.properties` est créé par le programme d'installation ou par la commande **`fteSetupCoordination`** . Vous pouvez utiliser la commande **`fteSetupCoordination`** avec l'indicateur **`-f`** pour modifier les propriétés de base du gestionnaire de files d'attente de coordination

dans ce fichier. Pour modifier ou ajouter des propriétés de gestionnaire de files d'attente de coordination avancées, vous devez éditer le fichier dans un éditeur de texte.

Le fichier `coordination.properties` se trouve dans votre répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name`.

Le fichier `coordination.properties` contient les valeurs suivantes:

<i>Tableau 40. Propriétés du gestionnaire de files d'attente de coordination</i>		
Nom de la propriété	Description	Valeur par défaut
coordinationQMgr	Nom du gestionnaire de file d'attente de coordination.	Aucune valeur par défaut
Hôte coordinationQMgr	Nom d'hôte ou adresse IP du gestionnaire de file d'attente de coordination.	Aucune valeur par défaut
Port coordinationQMgr	Numéro de port utilisé pour les connexions client au gestionnaire de file d'attente de coordination.	1414
Canal coordinationQMgr	Nom de canal SVRCONN utilisé pour la connexion au gestionnaire de file d'attente de coordination.	SYSTEM.DEF.SVRCONN

Si vous ne spécifiez pas de valeur pour la propriété d'hôte `coordinationQMgr`, le mode de liaison est utilisé par défaut.

Si vous spécifiez une valeur pour la propriété d'hôte `coordinationQMgr` mais que vous ne spécifiez pas de valeurs pour les propriétés de port `coordinationQMgr` et de canal `coordinationQMgr`, le numéro de port 1414 et le canal `SYSTEM.DEF.SVRCONN` sont utilisés par défaut.

<i>Tableau 41. Propriétés avancées du gestionnaire de files d'attente de coordination</i>		
Nom de la propriété	Description	Valeur par défaut
Propriétés de l'agent:		

Tableau 41. Propriétés avancées du gestionnaire de files d'attente de coordination (suite)

Nom de la propriété	Description	Valeur par défaut
agentStatusJitterTolerance	<p>Durée maximale pendant laquelle une publication de message d'état d'agent peut être retardée avant que le message ne soit considéré comme étant en retard. Cette valeur est mesurée en millisecondes.</p> <p>L'âge d'un message de statut dépend de l'heure à laquelle il a été publié au niveau du gestionnaire de files d'attente de coordination. Toutefois, le message est émis par l'agent un certain temps avant d'être reçu au niveau du gestionnaire de file d'attente de coordination afin de permettre le temps nécessaire pour parcourir le réseau IBM MQ . Si ce transit prend toujours le même temps, les messages créés à 60 secondes d'intervalle sont publiés à 60 secondes d'intervalle, quelle que soit la durée réelle du transit. Cependant, si le temps de transit varie entre les messages, ils peuvent être créés à des intervalles de 60 secondes mais publiés à des intervalles de 61, 59, 58 et 62 secondes par exemple. L'écart maximal de 60, 2 secondes dans cet exemple, est la gigue. Cette propriété détermine le délai maximal dû à la gigue avant que le message soit traité comme étant en retard.</p>	3000
Propriétés de page de codes :		

Tableau 41. Propriétés avancées du gestionnaire de files d'attente de coordination (suite)

Nom de la propriété	Description	Valeur par défaut
coordinationCcsid	Page de codes avec laquelle les commandes se connectent au gestionnaire de file d'attente de coordination. Toutes les publications du gestionnaire de files d'attente de coordination effectuées par l'agent sont également effectuées à l'aide de cette page de codes. Si vous spécifiez une valeur pour <code>coordinationCcsid</code> , vous devez également spécifier une valeur pour <code>coordinationCcsid</code> .	1208
coordinationCcsidNom	Représentation Java de <code>coordinationCcsid</code> . Si vous spécifiez une valeur pour le nom <code>coordinationCcsid</code> , vous devez également spécifier une valeur pour <code>coordinationCcsid</code> .	UTF8
Propriétés de connexion :		
javaLibraryPath	Lors de la connexion à un gestionnaire de files d'attente en mode liaisons, Managed File Transfer doit avoir accès aux bibliothèques de liaisons IBM MQ Java . Par défaut, Managed File Transfer recherche les bibliothèques de liaisons dans l'emplacement par défaut défini par IBM MQ. Si les bibliothèques de liaisons se trouvent à un autre emplacement, utilisez cette propriété pour spécifier l'emplacement des bibliothèques de liaisons.	<code>MQ_INSTALLATION_PATH/java/lib</code>
Propriétés de gestionnaire de files d'attente multi-instance :		
coordinationQMgrde secours	Nom d'hôte et numéro de port utilisés pour les connexions client, au format IBM MQ CONNAME, pour l'instance de secours d'un gestionnaire de files d'attente de coordination multi-instance défini par la propriété <code>coordinationQMgr</code> . Exemple : <code>host_name(port_number)</code>	Aucune valeur par défaut
Propriétés de file d'attente :		

Tableau 41. Propriétés avancées du gestionnaire de files d'attente de coordination (suite)

Nom de la propriété	Description	Valeur par défaut
Préfixe dynamicQueue	<p>Cette propriété définit le préfixe IBM MQ à utiliser pour générer un nom de file d'attente temporaire.</p> <p>Le format de la propriété de préfixe dynamicQueuesuit le format de la zone DynamicQName de la structure IBM MQ MQOD. Pour plus d'informations, voir Création de files d'attente dynamiques.</p> <p>Vous pouvez également définir cette propriété dans le fichier <code>command.properties</code> si vous souhaitez utiliser un préfixe IBM MQ spécifique pour les files d'attente de réponses temporaires générées par des commandes nécessitant une réponse de l'agent.</p>	WMQFTE.*
modelQueueNom	<p>Cette propriété définit la file d'attente modèle IBM MQ à utiliser pour générer une file d'attente temporaire.</p> <p>Vous pouvez également définir cette propriété dans le fichier <code>command.properties</code> si vous souhaitez utiliser une file d'attente modèle IBM MQ spécifique pour les files d'attente de réponses temporaires générées par des commandes nécessitant une réponse de l'agent.</p> <p>Pour plus d'informations, voir la section «Le fichier MFT command.properties», à la page 559.</p>	SYSTEM.DEFAULT.MODEL.QUEUE
Propriétés de sécurité :		
userIdForClientConnect	<p>ID utilisateur transmis par l'intermédiaire des connexions client à IBM MQ. Si <i>java</i> est spécifié, le nom d'utilisateur signalé par la machine virtuelle Java est transmis dans le cadre de la demande de connexion IBM MQ . La valeur de cette propriété peut être None ou <i>java</i>.</p>	Aucun

Tableau 41. Propriétés avancées du gestionnaire de files d'attente de coordination (suite)

Nom de la propriété	Description	Valeur par défaut
coordinationQMgrAuthenticationCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification de connexion MQ pour la connexion au gestionnaire de files d'attente de coordination.	<p> Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir Création d'un fichier de données d'identification MFT</p> <p> Pour plus d'informations sur l'emplacement et les droits d'accès de ce fichier, voir Configuration de MQMFTCredentials.xml .</p> <p> Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir MFT et IBM MQ authentification de la connexion .</p>
Propriétés d'abonnement:		
Rubrique coordinationSubscription	<p>Utilisez cette propriété pour spécifier une rubrique autre que SYSTEM.FTE auquel s'abonner pour obtenir des publications sur le statut du réseau IBM MQ . Tous les outils sont toujours publiés dans SYSTEM.FTE , mais vous pouvez modifier votre topologie IBM MQ pour distribuer ces publications dans différentes rubriques en fonction de leur contenu. Vous pouvez ensuite utiliser cette fonction pour forcer les outils à s'abonner à l'une de ces autres rubriques.</p> <p>Pour IBM WebSphere MQ 7.5 et les groupes de correctifs ultérieurs, vous avez besoin d'un correctif temporaire pour l'APAR IC96850 pour que la propriété soit reconnue par IBM MQ Explorer et la commande ftelistmonitors .</p>	SYSTEM.FTE

Le texte suivant est un exemple de contenu d'un fichier coordination.properties .

```
coordinationQMgr=ERIS
coordinationQMgrHost=kuiper.example.com
```

ERIS est le nom d'un gestionnaire de files d'attente IBM MQ situé sur le système `kuiper.example.com`. Le gestionnaire de files d'attente ERIS est le gestionnaire de files d'attente auquel Managed File Transfer envoie les informations de journal.

Concepts associés

«Options de configuration de MFT sur Multiplatforms», à la page 58

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

«fteSetupCoordination», à la page 527

La commande **fteSetupCoordination** crée des fichiers de propriétés ainsi que le répertoire du gestionnaire de file d'attente de coordination pour Managed File Transfer.

«Propriétés SSL pour MFT», à la page 622

Utilisez SSL ou TLS avec IBM MQ et Managed File Transfer pour empêcher les connexions non autorisées entre les agents et les gestionnaires de files d'attente et pour chiffrer le trafic des messages entre les agents et les gestionnaires de files d'attente.

Le fichier MFT `command.properties`

Le fichier `command.properties` indique le gestionnaire de files d'attente de commandes auquel vous devez vous connecter lorsque vous émettez des commandes et les informations dont Managed File Transfer a besoin pour contacter ce gestionnaire de files d'attente.

Le fichier `command.properties` est créé par le programme d'installation ou par la commande **fteSetupCommands**. Vous pouvez utiliser la commande **fteSetupCommands** avec l'indicateur **-f** pour modifier les propriétés de base du gestionnaire de files d'attente de commandes dans ce fichier. Pour modifier ou ajouter des propriétés de gestionnaire de files d'attente de commandes avancées, vous devez éditer le fichier dans un éditeur de texte.

Certaines commandes Managed File Transfer se connectent au gestionnaire de files d'attente d'agent ou au gestionnaire de files d'attente de coordination à la place du gestionnaire de files d'attente de commandes. Pour plus d'informations sur les commandes qui se connectent à un gestionnaire de files d'attente, voir «Les commandes et processus MFT qui se connectent à quel gestionnaire de files d'attente», à la page 376.

Le fichier `command.properties` se trouve dans votre répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name`.

Le fichier `command.properties` contient les valeurs suivantes:

Nom de la propriété	Description	Valeur par défaut
<code>connectionQMGr</code>	Nom du gestionnaire de files d'attente utilisé pour la connexion au réseau IBM MQ.	Aucune valeur par défaut
Hôte <code>connectionQMGr</code>	Nom d'hôte ou adresse IP du gestionnaire de files d'attente de connexion.	Aucune valeur par défaut
<code>connectionQMGrPort</code>	Numéro de port utilisé pour la connexion au gestionnaire de files d'attente de connexion en mode client.	1414

Tableau 42. Propriétés de base du gestionnaire de files d'attente de commandes (suite)

Nom de la propriété	Description	Valeur par défaut
Canal connectionQMgr	Nom de canal SVRCONN utilisé pour la connexion au gestionnaire de files d'attente de connexion.	SYSTEM.DEF.SVRCONN

Si vous ne spécifiez pas de valeur pour la propriété d'hôte connectionQMgr, le mode de liaison est utilisé par défaut.

Si vous spécifiez une valeur pour la propriété d'hôte connectionQMgr mais que vous ne spécifiez pas de valeurs pour les propriétés de port connectionQMgr et de canal connectionQMgr, le numéro de port 1414 et le canal SYSTEM.DEF.SVRCONN sont utilisés par défaut.

Tableau 43. Propriétés avancées du gestionnaire de files d'attente de commandes

Nom de la propriété	Description	Valeur par défaut
Propriétés de page de codes :		
connectionCcsid	Page de codes avec laquelle les commandes se connectent au gestionnaire de files d'attente de commandes. Si vous spécifiez une valeur pour connectionCcsid, vous devez également spécifier une valeur pour le nom connectionCcsid.	1208
connectionCcsidNom	Représentation Java de connectionCcsid. Si vous spécifiez une valeur pour le nom connectionCcsid, vous devez également spécifier une valeur pour connectionCcsid.	UTF8
Propriétés de gestionnaire de files d'attente multi-instance :		
connectionQMgrde secours	Nom d'hôte et numéro de port utilisés pour les connexions client, au format IBM MQ CONNAME, pour l'instance de secours d'un gestionnaire de files d'attente de commandes multi-instance défini par la propriété connectionQMgr. Exemple : <i>host_name(port_number)</i>	Aucune valeur par défaut
Propriétés de sécurité :		
userIdForClientConnect	ID utilisateur transmis par l'intermédiaire des connexions client à IBM MQ. Si <i>java</i> est spécifié, le nom d'utilisateur signalé par la machine virtuelle Java est transmis dans le cadre de la demande de connexion IBM MQ. La valeur de cette propriété peut être None ou <i>java</i> .	Aucun

Tableau 43. Propriétés avancées du gestionnaire de files d'attente de commandes (suite)

Nom de la propriété	Description	Valeur par défaut
connectionQMGrAuthenticationCredentialsFichier	Chemin d'accès au fichier qui contient les données d'identification de connexion MQ pour la connexion au gestionnaire de files d'attente de commandes.	<p> Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir Création d'un fichier de données d'identification MFT</p> <p> Pour plus d'informations sur l'emplacement et les droits d'accès de ce fichier, voir Configuration de MQMFTCredentials.xml.</p> <p> Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir MFT et IBM MQ authentification de la connexion.</p>
Propriétés de file d'attente :		
Préfixe dynamicQueue	<p>Pour les commandes qui requièrent une réponse de l'agent, cette propriété définit le préfixe IBM MQ à utiliser pour générer le nom de la file d'attente de réponses temporaire.</p> <p>Le format de la propriété de préfixe dynamicQueue suit le format de la zone DynamicQName de la structure IBM MQ MQOD. Pour plus d'informations, voir Création de files d'attente dynamiques.</p> <p>Vous pouvez également définir cette propriété dans le fichier <code>coordination.properties</code> si vous souhaitez utiliser un préfixe IBM MQ spécifique pour les files d'attente temporaires générées par WMQFTE.</p>	WMQFTE.*

Tableau 43. Propriétés avancées du gestionnaire de files d'attente de commandes (suite)

Nom de la propriété	Description	Valeur par défaut
modelQueueNom	<p>Pour les commandes qui requièrent une réponse de l'agent, cette propriété définit la file d'attente modèle IBM MQ à utiliser pour la génération de la file d'attente de réponses temporaire.</p> <p>Vous pouvez également définir cette propriété dans le fichier <code>coordination.properties</code> si vous souhaitez utiliser une file d'attente modèle IBM MQ spécifique pour les files d'attente temporaires générées par WMQFTE. Pour plus d'informations, voir la section «Le fichier MFT coordination.properties», à la page 553.</p>	SYSTEM.DEFAULT.MODEL.QUEUE
Propriétés de connexion :		
javaLibraryPath	<p>Lors de la connexion à un gestionnaire de files d'attente en mode liaisons, Managed File Transfer doit avoir accès aux bibliothèques de liaisons IBM MQ Java . Par défaut, Managed File Transfer recherche les bibliothèques de liaisons dans l'emplacement par défaut défini par IBM MQ. Si les bibliothèques de liaisons se trouvent à un autre emplacement, utilisez cette propriété pour spécifier l'emplacement des bibliothèques de liaisons.</p>	/opt/mqm/java/lib

Tableau 43. Propriétés avancées du gestionnaire de files d'attente de commandes (suite)

Nom de la propriété	Description	Valeur par défaut
 legacyXMLMessageMQMDFormat	Les messages XML de commande Managed File Transfer sont désormais envoyés à une file d'attente avec une zone de format MQMD vide. Les versions précédentes du produit définissaient la zone au format MQMD sur MQSTR (chaîne de message texte). La définition de cette propriété sur true permet d'envoyer les messages XML de la commande Managed File Transfer à une file d'attente avec la zone de format MQMD de MQSTR. Si la zone au format MQMD est définie sur MQSTR, les messages XML des commandes Managed File Transfer risquent d'être corrompus si la conversion des données est activée pour des canaux du réseau MQ.	false
Propriétés d'agent		
 failCleanAgentWithNoArguments	Par défaut, la valeur de cette propriété est true, ce qui signifie que l'exécution de la commande fteCleanAgent échoue si seul le paramètre de nom d'agent est spécifié. La définition de la propriété sur false signifie que, si seul le paramètre de nom d'agent est défini, le comportement de la commande fteCleanAgent est équivalent à la spécification du paramètre -all .	conforme

Le texte suivant est un exemple de contenu d'un fichier `command.properties`.

```
connectionQMgr=PLUTO
connectionQMgrHost=kuiper.example.com
connectionQMgrPort=1930
connectionQMgrChannel=SYSTEM.DEF.SVRCONN
```

PLUTO est le nom d'un gestionnaire de files d'attente IBM MQ qui se trouve sur le système `kuiper.example.com`. Le gestionnaire de files d'attente PLUTO est le gestionnaire de files d'attente auquel les commandes Managed File Transfer se connectent.

Concepts associés

«Options de configuration de MFT sur Multiplatforms», à la page 58

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

«Propriétés système Java pour MFT», à la page 621

Un certain nombre de propriétés de commande et d'agent Managed File Transfer doivent être définies en tant que propriétés système Java, car elles définissent la configuration de la fonction précoce qui ne peut pas utiliser le mécanisme des propriétés de commande ou d'agent.

«Propriétés SSL pour MFT», à la page 622

Utilisez SSL ou TLS avec IBM MQ et Managed File Transfer pour empêcher les connexions non autorisées entre les agents et les gestionnaires de files d'attente et pour chiffrer le trafic des messages entre les agents et les gestionnaires de files d'attente.

«Commandes fteSetup: créez le fichier MFT command.properties», à la page 524

La commande **fteSetupCommands** crée le fichier Managed File Transfer `command.properties`. Ce fichier de propriétés spécifie les détails du gestionnaire de files d'attente qui se connecte au réseau IBM MQ lorsque vous émettez des commandes.

«Agent fteClean: nettoyage d'un agent MFT», à la page 394

La commande **fteCleanAgent** permet de nettoyer les files d'attente utilisées par un Managed File Transfer Agent en supprimant les messages des files d'attente persistantes et non persistantes utilisées par l'agent. Utilisez la commande **fteCleanAgent** si vous rencontrez des problèmes lors du démarrage d'un agent, ce qui peut être dû aux informations restant dans les files d'attente utilisées par l'agent.

Le fichier MFT agent.properties

Chaque Managed File Transfer Agent possède son propre fichier de propriétés, `agent.properties`, qui doit contenir les informations qu'un agent utilise pour se connecter à son gestionnaire de files d'attente. Le fichier `agent.properties` peut également contenir des propriétés qui modifient le comportement de l'agent.

Le fichier `agent.properties` est créé par le programme d'installation ou par la commande **fteCreateAgent**, **fteCreateBridgeAgent** ou **fteCreateCDAgent**. Vous pouvez utiliser l'une de ces commandes avec l'option **-f** pour modifier les propriétés de base du gestionnaire de files d'attente de l'agent, ainsi que les propriétés d'agent avancées associées au type d'agent que vous créez. Pour modifier ou ajouter des propriétés d'agent avancées, vous devez éditer le fichier dans un éditeur de texte.

Le fichier `agent.properties` d'un agent se trouve dans votre répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name`.

z/OS Sous z/OS, l'emplacement du fichier `agent.properties` est `$BFG_CONFIG variable/mqft/config/coordination_qmgr_name/agents/agent_name`.

Si vous modifiez le fichier `agent.properties`, vous devez redémarrer l'agent pour appliquer vos modifications.

Pour IBM WebSphere MQ 7.5 et les versions ultérieures, vous pouvez utiliser des variables d'environnement dans certaines propriétés de Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela vous permet d'utiliser les emplacements de fichiers ou de répertoires lors de l'exécution de composants du produit en fonction des modifications d'environnement (par exemple, en fonction de l'utilisateur qui exécute le processus). Pour plus d'informations, voir la section «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.

Propriétés de l'agent de base

Chaque fichier `agent.properties` contient les propriétés de base suivantes.

Nom de la propriété	Description	Valeur par défaut
agentName	Nom de l'agent. Le nom de l'agent doit être conforme aux conventions de dénomination des objets IBM MQ. Pour plus d'informations, voir la section «Conventions de dénomination des objets MFT», à la page 641.	Aucune valeur par défaut

Tableau 44. Propriétés de l'agent de base (suite)

Nom de la propriété	Description	Valeur par défaut
agentDesc	Description de l'agent (si vous choisissez de créer une description).	Aucune valeur par défaut
agentQMgr	Nom du gestionnaire de files d'attente de l'agent.	Aucune valeur par défaut
agentQMgrHost	Nom d'hôte ou adresse IP du gestionnaire de files d'attente de l'agent.	Aucune valeur par défaut
agentQMgrPort	Numéro de port utilisé pour établir les connexions client avec le gestionnaire de files d'attente de l'agent.	1414
agentQMgrChannel	Nom de canal SVRCONN utilisé pour se connecter au gestionnaire de files d'attente de l'agent.	SYSTEM.DEF.SVRCONN
agentType	Type d'agent : <ul style="list-style-type: none"> • Agent standard autre qu'un agent de pont (STANDARD) • Agent de pont de protocole (BRIDGE) • Agent de pont Connect:Direct (CD_BRIDGE) • Agent imbriqué tel qu'il est utilisé par IBM Integration Bus (EMBEDED) • Agent imbriqué à SFG (Sterling File Gateway) 	Standard

Si vous ne spécifiez pas de valeur pour la propriété agentQMgrHost, le mode liaisons est utilisé par défaut.

Si vous spécifiez une valeur pour la propriété agentQMgrHost, mais que vous n'en spécifiez pas pour les propriétés agentQMgrPort et agentQMgrChannel, le numéro de port 1414 et le canal SYSTEM.DEF.SVRCONN sont utilisés par défaut.

Propriétés d'agent avancées

Managed File Transfer fournit également davantage de propriétés d'agent avancées qui vous aident à configurer les agents. Si vous souhaitez utiliser l'une des propriétés ci-après, éditez manuellement le fichier `agent.properties` pour ajouter les propriétés avancées requises. Lorsque vous spécifiez des chemins de fichier sous Windows, assurez-vous que la barre oblique inversée (\) du caractère de séparation est entrée en tant que double barre oblique inversée (\\), c'est-à-dire que la barre oblique inversée (\) est mise en échappement. Vous pouvez également utiliser une barre oblique (/) comme séparateur. Pour plus d'informations sur l'échappement des caractères dans les fichiers de propriétés Java, reportez-vous à la rubrique [Javadoc for the Properties class](#) dans la documentation Oracle.

- [Propriétés de taille d'agent](#)
- [Propriétés de page de codes](#)
- [Propriétés de commande](#)
- [Propriétés de connexion](#)
- [Propriétés de pontConnect:Direct](#)
- [Propriétés d'agent de fichier à message et de message à fichier](#)
- [Propriétés d'agent générales](#)
- [Propriétés d'entrée/de sortie](#)
- [Propriétés de prise en charge multicanal](#)
- [Propriétés multiinstance](#)
- [Propriétés de contrôleur de processus](#)

- Propriétés de pont de protocole
- **V 9.0.0** Propriétés de journal d'agent de pont de protocole
- Propriétés de file d'attente
- Propriétés de surveillance des ressources
- Propriétés du répertoire racine
- Propriété Scheduler
- Propriétés de sécurité
- Propriétés de délai d'expiration
- **V 9.0.1** Propriétés de délai d'expiration pour la reprise des transferts
- Propriétés de trace et de consignation
- Propriétés de limite de transfert
- Propriétés de routine d'exit utilisateur
- Propriétés de compression du client IBM MQ
- **z/OS** Propriétés spécifiques à z/OS
- Autres propriétés

Tableau 45. Propriétés d'agent avancées

Nom de la propriété	Description	Valeur par défaut
Propriétés de taille d'agent :		
agentCheckpointInterval	<p>Intervalle en cadres de données complets au cours duquel un point de contrôle est effectué à des fins de récupération. Il s'agit d'une propriété avancée et pour la plupart des configurations Managed File Transfer, il n'est pas nécessaire d'en modifier la valeur.</p> <p>En cas de problème entraînant le passage du transfert en récupération, le transfert ne peut être restauré que sur une limite de point de contrôle. Par conséquent, plus cette valeur est élevée (avec des valeurs agentChunkSize, agentWindowSize et agentFrameSize élevées), plus l'agent a besoin de temps pour récupérer les transferts. Pour les réseaux Managed File Transfer fiables où les transferts ne passent que rarement à l'état de récupération, il peut être judicieux d'augmenter cette valeur pour augmenter les performances globales.</p>	1
agentChunkSize	<p>Taille de chaque bloc de transfert pour le transport des données de fichier. Indique donc la taille maximale des messages IBM MQ transférés entre les agents source et de destination. Il s'agit d'une propriété avancée et pour la plupart des configurations Managed File Transfer, il n'est pas nécessaire d'en modifier la valeur.</p> <p>Cette valeur est négociée entre les agents source et de destination, la valeur la plus élevée des deux étant utilisée. Si vous souhaitez modifier la valeur de cette propriété, remplacez-la au niveau de l'agent source et de l'agent de destination.</p> <p>agentChunkSize est une valeur entière. Par exemple: agentChunkSize = 10240 définit la taille de bloc sur 10 Ko.</p>	262144-byte (équivalent à 256 ko)
agentFrameSize	<p>Nombre de fenêtres pour le cadre de transfert. Il s'agit d'une propriété avancée et pour la plupart des configurations Managed File Transfer, il n'est pas nécessaire d'en modifier la valeur.</p> <p>Pour les réseaux dont le temps d'attente est élevé, l'augmentation de cette valeur peut améliorer les performances globales car elle permet à l'agent d'avoir plus de blocs de message actifs simultanément.</p> <p>La valeur de cette propriété, multipliée par celle d'agentWindowSize, multipliée par celle d'agentChunkSize, indique la limite supérieure de la consommation de mémoire de l'agent pour chaque transfert. Par exemple, blocs de 262 144 octets x 10 x 5 = 12,5 Mo par transfert.</p> <p>Remarque : si la taille des fichiers transférés au cours d'un même transfert est inférieure à 12,5 Mo, l'augmentation de la valeur de cette propriété n'a aucune incidence sur les performances du transfert.</p>	5

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
agentWindowSize	<p>Nombre de blocs de chaque fenêtre. Il s'agit d'une propriété avancée et pour la plupart des configurations Managed File Transfer, il n'est pas nécessaire d'en modifier la valeur.</p> <p>Pour les réseaux dont le temps d'attente est élevé, l'augmentation de cette valeur peut améliorer les performances globales. Cela est dû au fait que l'agent a plus de blocs de messages actifs simultanément et réduit la fréquence à laquelle les messages d'accusé de réception sont renvoyés à l'agent source.</p> <p>La valeur de cette propriété, multipliée par celle d'agentFrameSize, multipliée par celle d'agentChunkSize, indique la limite supérieure de la consommation de mémoire de l'agent pour chaque transfert et la limite supérieure des données de message IBM MQ sur la file d'attente des commandes de l'agent de destination. Par exemple, blocs de 262 144 octets x 10 x 5 = limite supérieure de 12,5 Mo par transfert.</p> <p>Remarque : si la taille des fichiers transférés au cours d'un même transfert est inférieure à 12,5 Mo, l'augmentation de la valeur de cette propriété n'a aucune incidence sur les performances du transfert.</p>	10
Propriétés de page de codes :		
agentCcsid	Page de codes avec laquelle l'agent se connecte à son gestionnaire de files d'attente d'agent. Si vous spécifiez une valeur pour agentCcsid, vous devez également en spécifier une pour agentCcsidName. Pour plus d'informations sur la manière d'afficher les pages de codes connues pour la machine virtuelle Java, reportez-vous au paramètre -hsc de la commande ftCreateBridgeAgent .	1208
agentCcsidName	Représentation Java d'agentCcsid. Si vous spécifiez une valeur pour agentCcsidName, vous devez également en spécifier une pour agentCcsid.	UTF8
Propriétés de commande :		
maxCommandHandlerThreads	Contrôle le nombre d'unités d'exécution disponibles pour l'analyse initiale et le traitement des messages de commande de transfert. Lorsqu'elles sont actives, les unités d'exécution requièrent une connexion au gestionnaire de files d'attente, mais elles la libèrent lorsqu'elles sont inactives.	5
maxCommandOutput	Nombre maximal d'octets stockés pour la sortie de commande. Cette propriété s'applique aux commandes spécifiées pour un appel géré et aux commandes preSource, postSource, preDestination et postDestination d'un transfert géré. Cela permet de limiter la longueur de la sortie de commande consignée dans le journal de transfert sur la rubrique SYSTEM.FTE.	10240
maxCommandRetries	Nombre maximal de nouvelles tentatives d'une commande autorisées par l'agent. Cette propriété s'applique aux commandes spécifiées pour un appel géré et aux commandes preSource, postSource, preDestination et postDestination d'un transfert géré.	9
maxCommandWait	Délai d'attente maximal, en secondes, entre les nouvelles tentatives autorisées par l'agent. Cette propriété s'applique aux commandes spécifiées pour un appel géré et aux commandes preSource, postSource, preDestination et postDestination d'un transfert géré.	60
immediateShutdownTimeout	<p>Pour un arrêt immédiat d'un agent, vous pouvez utiliser cette propriété pour indiquer la durée maximale en secondes pendant laquelle un agent attend la fin de ses transferts avant de forcer un arrêt.</p> <p>Remarque : Ne spécifiez pas une valeur inférieure à la valeur par défaut de 10 secondes pour cette propriété. Un arrêt immédiat d'un agent requiert suffisamment de temps pour arrêter les éventuels processus externes. Si la valeur de cette propriété est trop faible, des processus peuvent rester actifs.</p> <p>Si la valeur 0 est spécifiée pour cette propriété, l'agent attend que tous les transferts en attente soient arrêtés. Si une valeur non valide est spécifiée pour cette propriété, la valeur par défaut est utilisée.</p>	10
Propriétés de connexion :		
javaLibraryPath	Lors de la connexion à un gestionnaire de files d'attente en mode liaisons, Managed File Transfer doit avoir accès aux bibliothèques de liaisons IBM MQ Java . Par défaut, Managed File Transfer recherche ces bibliothèques de liaisons dans l'emplacement par défaut défini par IBM MQ. Si elles se trouvent dans un autre emplacement, utilisez cette propriété pour en spécifier l'emplacement.	Aucun
Propriétés de pont Connect:Direct :		

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
cdNode	Propriété requise si vous souhaitez utiliser le pont Connect:Direct. Nom du noeud Connect:Direct à utiliser pour transférer les messages de l'agent de pont Connect:Direct vers les noeuds Connect:Direct de destination. Ce noeud fait partie du pont Connect:Direct et non du noeud distant qui correspond à la source ou à la destination du transfert. Pour plus d'informations, voir la section «Le pont Connect:Direct», à la page 236.	Aucune valeur par défaut
cdNodeHost	Nom d'hôte ou adresse IP du noeud Connect:Direct à utiliser pour transférer les fichiers de l'agent de pont Connect:Direct vers les noeuds de destination (noeud de pont Connect:Direct). Dans la plupart des cas, le noeud de pont Connect:Direct se trouve sur le même système que l'agent de pont Connect:Direct. Dans ces cas là, la valeur par défaut de cette propriété (l'adresse IP du système local) est correcte. Si votre système possède plusieurs adresses IP ou que votre noeud de pont Connect:Direct se trouve sur un autre système que votre agent de pont Connect:Direct et que leurs systèmes partagent un système de fichiers, utilisez cette propriété pour spécifier le nom d'hôte correct du noeud de pont Connect:Direct. Si vous n'avez pas défini la propriété cdNode, cette propriété est ignorée.	Nom d'hôte ou adresse IP du système local
cdNodePort	Numéro de port du noeud de pont Connect:Direct utilisé par les applications client pour communiquer avec le noeud. Dans la documentation du produit Connect:Direct, ce port est appelé port d'API. Si vous n'avez pas défini la propriété cdNode, cette propriété est ignorée.	1363
cdTmpDir	Emplacement permettant de stocker les fichiers temporairement sur le système où l'agent de pont Connect:Direct est exécuté avant qu'ils ne soient transférés vers le noeud Connect:Direct de destination. Cette propriété spécifie le chemin d'accès complet du répertoire dans lequel les fichiers sont stockés temporairement. Par exemple, si cdTmpDir est défini sur /tmp, les fichiers sont temporairement placés dans le répertoire /tmp. Si vous n'avez pas défini la propriété cdNode, cette propriété est ignorée. La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.	<code>value_of_java.io.tmpdir/cdbridge-agentName</code> Sous Windows : <code>value_of_java.io.tmpdir\cdbridge-agentName</code>
cdTrace	Indique si l'agent trace les données envoyées entre l'agent de pont Connect:Direct et son noeud Connect:Direct. La valeur de cette propriété peut être <code>true</code> ou <code>false</code> .	false
cdMaxConnectionRetries	Nombre maximal de tentatives de connexion Connect:Direct, pour un transfert de fichiers où aucune connexion n'a encore été établie, avant que le transfert n'échoue.	-1 (nombre infini de tentatives)
cdMaxPartialWorkConnectionRetries	Nombre maximal de tentatives de connexion Connect:Direct, pour un transfert de fichiers où une tentative de connexion précédente et le travail de transfert ont abouti, avant que le transfert n'échoue.	-1 (nombre infini de tentatives)
cdMaxWaitForProcessEndStats	Délai d'attente maximal en millisecondes avant que les informations sur l'achèvement du processus Connect:Direct ne deviennent disponibles dans les statistiques du noeud Connect:Direct, une fois le processus terminé et avant que le transfert de fichier ne soit considéré comme ayant échoué. En général, ces informations sont disponibles immédiatement, mais sous certaines conditions d'échec, elles ne sont pas publiées. Dans ces conditions, le transfert de fichier échoue après le délai d'attente spécifié par cette propriété.	60 000
cdAppName	Nom d'application utilisé par l'agent de pont Connect:Direct pour se connecter au noeud Connect:Direct qui fait partie du pont.	Managed File Transfer <i>version actuelle</i> , <i>version actuelle</i> représentant le numéro de version du produit.
cdNodeLocalPortRange	Plage de ports locaux à utiliser pour les connexion au socket entre l'agent de pont Connect:Direct et le noeud Connect:Direct qui fait partie du pont. Le format de cette valeur est une liste de valeurs ou de plages, séparées par des virgules. Par défaut, le système d'exploitation choisit les numéros de port locaux.	Aucun
cdNodeProtocol	Protocole utilisé par l'agent de pont Connect:Direct pour se connecter au noeud Connect:Direct qui fait partie du pont. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> TCP/IP SSL TLS 	TCP/IP

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
cdNodeKeystore	Chemin d'accès au magasin de clés utilisé pour les communications sécurisées entre l'agent de pont Connect:Direct et le noeud Connect:Direct qui fait partie du pont. Si vous n'avez pas défini la propriété cdNodeProtocol sur SSL ou TLS, cette propriété est ignorée. La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.	Aucun
cdNodeKeystoreType	Format de fichier du magasin de clés spécifié par la propriété cdNodeKeystore. Les valeurs suivantes sont valides : jks et pkcs12. Si vous n'avez pas défini la propriété cdNodeProtocol sur SSL ou TLS, cette propriété est ignorée.	jks
cdNodeKeystoreCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification de cdNodeKeystore. La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.	 Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir Création d'un fichier de données d'identification MFT  Pour plus d'informations sur l'emplacement et les droits d'accès de ce fichier, voir Configuration de MQMFTCredentials.xml .  Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir MFT et IBM MQ authentification de la connexion .
cdNodeTruststore	Chemin d'accès au fichier de clés certifiées utilisé pour les communications sécurisées entre l'agent de pont Connect:Direct et le noeud Connect:Direct qui fait partie du pont. Si vous n'avez pas défini la propriété cdNodeProtocol sur SSL ou TLS, cette propriété est ignorée. La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.	Aucun
cdNodeTruststoreType	Format de fichier du fichier de clés certifiées spécifié par la propriété cdNodeTruststore. Les valeurs suivantes sont valides : jks et pkcs12. Si vous n'avez pas défini la propriété cdNodeProtocol sur SSL ou TLS, cette propriété est ignorée.	jks
cdNodeTruststoreCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification de cdNodeTruststore. La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.	Voir Création d'un fichier de données d'identification MFT et Configuration de MQMFTCredentials.xml  Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir Création d'un fichier de données d'identification MFT .  Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir MFT et IBM MQ authentification de la connexion .
logCDProcess	Niveau de consignation des processus Connect:Direct enregistré dans le journal des événements d'agent, dans le fichier output0.log. Les valeurs possibles pour cette propriété sont None, Failures et All.	Aucun

Propriétés d'agent de fichier à message et de message à fichier :

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
deleteTmpFileAfterRenameFailure	Si vous définissez cette propriété sur false, les fichiers temporaires ne sont pas supprimés de la destination en cas d'échec de l'opération de changement de nom. Dans ce cas, les données transférées restent sur la destination, dans un fichier temporaire (.part). Vous pourrez renommer manuellement ce fichier ultérieurement. Par défaut, cette propriété possède la valeur true. Cette propriété s'applique à la fois aux transferts de message à fichier et de fichier à fichier.	conforme
enableQueueInputOutput	Par défaut, l'agent ne peut pas lire les données d'une file d'attente source ou écrire les données dans une file d'attente de destination dans le cadre d'un transfert. La définition de cette valeur sur true permet à l'agent d'effectuer des transferts de fichier à message et de message à fichier. La valeur de cette propriété peut être true ou false.	false
enableSystemQueueInputOutput	Indique si l'agent peut lire les données des files d'attente système d'IBM MQ ou écrire dessus. Les files d'attente système sont préfixées du qualificatif SYSTEM. Remarque : les files d'attente système sont utilisées par IBM MQ, Managed File Transfer et d'autres applications pour transmettre des informations importantes. Si vous changez cette propriété, l'agent peut accéder à ces files d'attente. Si vous activez cette propriété, utilisez les bacs à sable utilisateur pour limiter les files d'attente accessibles par l'agent.	false
enableClusterQueueInputOutput	Indique si l'agent peut lire les données des files d'attente en cluster d'IBM MQ ou écrire dessus. Remarque : Vous devez spécifier la propriété d'agent enableClusterQueueInputOutput en plus de la propriété enableQueueInputOutput.	false
maxDelimiterMatchLength	Nombre maximal de caractères pouvant correspondre avec l'expression régulière Java utilisée pour fractionner un fichier texte en plusieurs messages dans le cadre d'un transfert de fichier à message.	5
maxInputOutputMessageLength	Longueur maximale, en octets, d'un message lu à partir d'une file d'attente source ou écrit dans une file d'attente de destination par un agent. La propriété maxInputOutputMessageLength de l'agent source dans un transfert détermine le nombre d'octets pouvant être lus à partir d'un message sur la file d'attente source. La propriété maxInputOutputMessageLength de l'agent de destination dans un transfert détermine le nombre d'octets pouvant être écrits dans un message sur la file d'attente de destination. Si la longueur de ce message est supérieure à la valeur de cette propriété, le transfert échoue avec une erreur. Cette propriété n'affecte pas les files d'attente Managed File Transfer internes. Pour plus d'informations sur la modification de cette propriété, voir «Conseils pour la définition des attributs MQ et des propriétés MFT associées à la taille des messages», à la page 329.	1048576
monitorGroupRetryLimit	Nombre maximal de fois qu'un moniteur déclenche à nouveau un transfert de message à fichier si le groupe de messages existe toujours dans la file d'attente. Le nombre de fois que le transfert de message à fichier est déclenché est déterminé par le nombre d'annulations MQMD du premier message dans le groupe. Si l'agent est redémarré, le moniteur déclenche à nouveau un transfert, même si le nombre de déclenchements de transfert dépasse la valeur de monitorGroupRetryLimit. Si, en raison de ce comportement, le nombre de fois que le transfert est déclenché dépasse la valeur de monitorGroupRetryLimit, l'agent consigne une erreur dans son journal des événements. Si la valeur -1 est spécifiée pour cette propriété, le moniteur déclenche à nouveau le transfert un nombre illimité de fois, jusqu'à ce que la condition de déclenchement ne soit pas satisfaite.	10
Propriétés d'agent générales :		
agentStatusPublishRateLimit	Fréquence maximale, en secondes, à laquelle l'agent republie son statut en raison d'un changement du statut du transfert de fichier. Si vous affectez à cette propriété une valeur trop faible, cela peut avoir un impact négatif sur les performances du réseau IBM MQ.	30
agentStatusPublishRateMin	Fréquence minimale, en secondes, à laquelle l'agent publie son statut. Cette valeur doit être supérieure ou égale à la valeur de la propriété agentStatusPublishRateLimit.	300

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
enableMemoryAllocationChecking	<p>Détermine si l'Managed File Transfer Agent vérifie que la quantité de mémoire disponible est suffisante pour exécuter un transfert avant d'accepter ce dernier. La vérification est effectuée à la fois sur l'agent source et sur l'agent de destination. Si la quantité de mémoire disponible est insuffisante, le transfert est rejeté.</p> <p>Lors du calcul de la quantité de mémoire requise pour un transfert, la quantité de mémoire maximale requise par le transfert est utilisée. Par conséquent, la valeur peut être supérieure à la quantité de mémoire réelle utilisée par le transfert. Pour cette raison, le nombre de transferts simultanés pouvant être exécutés peut être inférieur si la propriété enableMemoryAllocationChecking est définie sur true. Il est recommandé de ne définir cette valeur sur true que si Managed File Transfer échoue avec des erreurs de mémoire insuffisante. Les transferts susceptibles de consommer des quantités importantes de mémoire sont les transferts de fichier à message et de message à fichier avec des messages de grande taille.</p>	false
enableDetailedReplyMessages	<p>La définition de cette propriété sur true permet aux réponses aux demandes de transfert gérées de contenir des informations détaillées sur les fichiers transférés. Les informations détaillées et le format sont identiques à ceux publiés dans le journal de transfert dans les messages de progression, c'est-à-dire l'élément <transferSet>. Pour plus d'informations, voir la section «Formats des messages du journal de transfert de fichiers», à la page 771.</p> <p>Les informations de réponse détaillées ne sont incluses que si la demande de transfert gérée indique que les informations de réponse détaillées sont requises. Pour spécifier cette exigence, définissez l'attribut detailed de l'élément <reply> du message de demande XML managedTransfer envoyé à l'agent source. Pour plus d'informations, voir la section «Format de message de demande de transfert de fichier», à la page 804.</p> <p>Il est possible de générer plusieurs messages de réponse par demande de transfert. Ce nombre est égal au nombre de messages de progression du journal de transfert pour le transfert, plus 1 (le premier message de réponse correspondant à une réponse d'accusé de réception simple). Les informations détaillées sont incluses dans tous les messages, exceptés les messages de réponse d'accusé de réception, mais le résultat global du transfert n'est inclus que dans le dernier message de réponse détaillé.</p>	conforme
enableUserMetadataOptions	<p>Détermine si vous pouvez utiliser des clés connues pour les métadonnées définies par l'utilisateur dans les nouvelles demandes de transfert pour fournir davantage d'options de transfert. Ces clés connues commencent toujours par le préfixe com.ibm.wmqfte.. Par conséquent, si la propriété enableUserMetadataOptions est définie sur true, les clés qui utilisent ce préfixe ne sont pas prises en charge pour une utilisation définie par l'utilisateur. Si la propriété enableUserMetadataOptions est définie sur true, les clés actuellement prises en charge sont les suivantes :</p> <ul style="list-style-type: none"> • com.ibm.wmqfte.insertRecordLineSeparator • com.ibm.wmqfte.newRecordOnLineSeparator • com.ibm.wmqfte.convertLineSeparators <p>Pour plus d'informations sur la signification de ces clés, voir «fteCreateTransfer: démarrer un nouveau transfert de fichier», à la page 447.</p> <p>La valeur de cette propriété peut être true ou false.</p>	false

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
failTransferOnFirstEchec	<p>Permet à un agent d'être configuré pour faire échouer un transfert géré dès qu'un élément de transfert dans ce transfert géré échoue.</p> <p>Pour activer cette fonction, l' APAR IT03450 doit être appliqué à la fois à l'agent source et à l'agent cible, et la propriété failTransferOnFirstFailure doit être définie sur true dans le fichier agent . properties de l'agent source. La définition de la propriété sur true true sur l'agent de destination est facultative.</p> <p>Lorsque la propriété failTransferOnFirstFailure est définie sur true, l'agent démarre le traitement normal des demandes de transfert gérées. Toutefois, dès qu'un élément de transfert échoue, le transfert géré est marqué comme ayant échoué et aucun autre élément de transfert n'est traité. Les éléments de transfert qui ont été traités avec succès avant l'échec du transfert géré sont traités de la manière suivante:</p> <ul style="list-style-type: none"> • La disposition source de ces éléments de transfert est respectée. Par exemple, si la disposition source de l'élément de transfert a été définie sur delete, le fichier source est supprimé. • Les fichiers de destination qui ont été écrits restent sur le système de fichiers de destination et ne sont pas supprimés. <p>Si la propriété failTransferOnFirstFailure n'est pas définie sur true et qu'un transfert de fichier géré contient plusieurs fichiers et que l'un de ces fichiers ne parvient pas à être transféré, par exemple parce que le fichier de destination existe déjà et que la propriété d'écrasement est définie sur exist, l'agent source continue et tente de transférer les fichiers restants dans la demande.</p>	false
itemsPerProgressMessage	<p>Nombre de fichiers transférés avant qu'un agent ne publie son prochain message du journal de progression. Utilisez cette propriété pour contrôler la fréquence de publication des messages du journal de progression dans le gestionnaire de file d'attente de coordination lors d'un transfert.</p> <p>La valeur maximale sur laquelle cette propriété peut être définie est de 1000.</p> <p>Remarque : Les messages de progression incluent des informations sur chacun des fichiers transférés depuis que le dernier message de progression a été publié. L'augmentation de cette valeur augmente la taille des messages de progression, ce qui peut avoir une incidence sur les performances.</p>	50
maxInlineFileSize	<p>Pour les transferts de fichier à fichier ou de fichier à message, taille de fichier maximale (en octets) pouvant être incluse automatiquement dans le message de demande de transfert initial.</p> <p>Vous pouvez utiliser cette propriété pour accélérer vos transferts, mais si vous spécifiez une taille de fichier trop élevée, cela peut entraîner une dégradation des performances. Il est conseillé d'utiliser une taille initiale de 100 ko pour cette propriété, mais il est recommandé de tester rigoureusement différentes valeurs jusqu'à ce que vous trouviez la taille de fichier qui convient le mieux pour votre système.</p>	0
Propriétés d'entrée/de sortie :		
doNotUseTempOutputFile	<p>Par défaut, l'agent écrit dans un fichier temporaire sur la destination et renomme ce fichier temporaire avec le nom de fichier requis une fois le transfert de fichier terminé. Si vous définissez cette valeur sur true, l'agent écrit directement dans le fichier de destination final.</p> <p> Sur les systèmes z/OS, ce comportement ne s'applique pas aux fichiers séquentiels, mais il s'applique aux membres des fichiers PDS.</p> <p>La valeur de cette propriété pour un transfert est définie par l'agent de destination.</p>	false

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
enableMandatoryLocking	<p>Lors de l'accès à des fichiers normaux, Managed File Transfer utilise un verrou partagé pour la lecture et un verrou exclusif pour l'écriture.</p> <p>Windows Sur Windows, le verrouillage de fichier est uniquement une recommandation. Si cette propriété est définie sur true, Managed File Transfer applique le verrouillage des fichiers. Sous Windows, cela signifie que si un fichier est ouvert dans une autre application, la surveillance de ce fichier n'est pas déclenchée tant qu'il n'est pas fermé. Les transferts Managed File Transfer impliquant ce fichier échouent.</p> <p>UNIX Sur les plateformes de type UNIX, le verrouillage des fichiers est réalisé entre les processus. Pour les plateformes de type UNIX, la définition de cette propriété n'a aucune incidence.</p> <p>Cette propriété s'applique uniquement aux agents Managed File Transfer normaux. Managed File Transfer ne prend pas en charge le mécanisme de verrouillage de fichier sur un serveur de fichiers. Cette propriété ne fonctionne donc pas pour un agent de pont de protocole car l'agent de pont de protocole ne verrouille pas un fichier sur un serveur de fichiers lors du transfert d'un fichier.</p> <p>La valeur de cette propriété peut être true ou false.</p>	false
ioIdleThreadTimeout	Durée en millisecondes pendant laquelle une unité d'exécution d'entrée/de sortie du système de fichiers doit rester inactive avant d'être arrêtée.	10000
ioQueueDepth	Nombre maximal de demandes d'entrée/de sortie à placer en file d'attente.	10
ioThreadPoolSize	<p>Nombre maximal d'unités d'exécution d'entrée/de sortie du système de fichiers disponibles. En général, chaque transfert utilise sa propre unité d'exécution d'entrée/de sortie de système de fichiers, mais si le nombre de transferts simultanés dépasse cette limite, les unités d'exécution d'entrée/de sortie de système de fichiers sont partagées entre les transferts.</p> <p>Si vous pensez avoir régulièrement plus de transferts simultanés en cours que la valeur ioThreadPoolSize, vous pouvez constater une amélioration en augmentant cette valeur, afin que chaque transfert possède sa propre unité d'exécution d'entrée/de sortie de système de fichiers.</p>	10
textReplacementCharacterSequence	<p>Pour un transfert en mode texte, si des octets de données ne peuvent pas être convertis de la page de codes source vers la page de codes de destination, le transfert de fichier échoue par défaut.</p> <p>Définissez cette propriété pour permettre au transfert de se terminer correctement en insérant la valeur de caractère spécifiée. Cette valeur de propriété est un caractère unique. Généralement, un point d'interrogation (?) est utilisé pour les caractères non mappables. Par exemple, utilisez le format textReplacementCharacterSequence=? où le point d'interrogation (?) est le caractère de remplacement. Vous ne pouvez pas utiliser d'espace blanc comme caractère de remplacement.</p>	Aucun
Prise en charge multicanal :		
agentMultipleChannelsEnabled	<p>La définition de cette propriété sur true permet à un Managed File Transfer Agent d'envoyer des messages de données de transfert sur plusieurs canaux IBM MQ. Dans certains scénarios, la définition de cette propriété peut améliorer les performances. Toutefois, n'activez la prise en charge multicanal que si son incidence positive sur les performances est prouvée. Seuls les messages placés dans la file d'attente SYSTEM.FTE.DATA.nomAgentDestination sont envoyés sur plusieurs canaux. Le comportement de tous les autres messages reste inchangé.</p> <p>Si vous définissez cette propriété sur true, vous devez également effectuer les étapes de configuration d'IBM MQ de l'une des rubriques suivantes pour activer la prise en charge multicanal :</p> <ul style="list-style-type: none"> « Configuration d'un agent MFT pour plusieurs canaux dans un cluster », à la page 585 « Configuration d'un agent MFT pour plusieurs canaux: non configurés en cluster », à la page 586 <p>Par ailleurs, vous devez également effectuer les étapes de configuration standard d'IBM MQ requises pour un agent Managed File Transfer, qui sont détaillées dans la rubrique « Configuration de MFT pour la première utilisation », à la page 97.</p> <p>La valeur de cette propriété peut être true ou false.</p>	false

Tableau 45. Propriétés d'agent avancées (suite)		
Nom de la propriété	Description	Valeur par défaut
agentMessageBatchSize	Si il est configuré avec plusieurs canaux, un agent source envoie les messages de données d'un transfert sur chaque canal, par permutation circulaire. Cette propriété contrôle le nombre de messages envoyés simultanément par chaque canal.	5
Propriétés de gestionnaire de files d'attente multi-instance :		
agentQMGrStandby	Nom d'hôte et numéro de port utilisés pour les connexions client, au format CONNAME d'IBM MQ, pour l'instance de secours d'un gestionnaire de files d'attente d'agent multi-instance défini par agentQMGr. Exemple : <i>host_name(port_number)</i> L'agent tente de se connecter au gestionnaire de files d'attente de secours lorsqu'il détecte une erreur de connexion interrompue, par exemple, MQRC 2009. Une fois que l'agent est connecté au gestionnaire de files d'attente de secours, il reste connecté jusqu'à ce que le gestionnaire de files d'attente de secours devienne indisponible.	Aucune valeur par défaut
Propriétés du contrôleur de processus :		
agentQMGrRetryInterval	Intervalle, en secondes, entre les vérifications de disponibilité du gestionnaire de files d'attente par le contrôleur de processus de l'agent.	30
maxRestartCount	Nombre maximal de redémarrages pouvant se produire dans l'intervalle de temps spécifié par la valeur de la propriété maxRestartInterval. Si cette valeur est dépassée, le contrôleur de processus de l'agent arrête le redémarrage de l'agent et effectue à la place une action qui dépend de la valeur de la propriété maxRestartDelay.	4
maxRestartInterval	Intervalle, en secondes, au cours duquel le contrôleur de processus de l'agent mesure les redémarrages de ce dernier. Si le nombre de redémarrages au cours de cet intervalle dépasse la valeur de la propriété maxRestartCount, le contrôleur de processus de l'agent arrête le redémarrage de l'agent. Au lieu de cela, il effectue une action qui dépend de la valeur de la propriété maxRestartDelay.	130
maxRestartDelay	Détermine le comportement du contrôleur de processus de l'agent si la fréquence des redémarrages de l'agent dépasse la valeur des propriétés maxRestartCount et maxRestartInterval. Si vous spécifiez une valeur inférieure ou égale à zéro, le contrôleur de processus de l'agent est arrêté. Si vous spécifiez une valeur supérieure à zéro, cette dernière correspond au nombre de secondes à attendre avant que les informations sur l'historique des redémarrages conservées par le contrôleur de processus de l'agent ne soient réinitialisées et que l'agent ne soit redémarré.	-1
Propriétés du pont de protocole :		
protocolBridgeCredentialConfiguration	La valeur de cette propriété est transmise comme chaîne à la méthode initialize() des classes d'exit spécifiées par protocolBridgeCredentialExitClasses.	null
protocolBridgeCredentialExitClasses	Spécifie une liste de classes implémentant une routine d'exit utilisateur des données d'identification du pont de protocole. Pour plus d'informations, voir la section «Mappage des données d'identification pour un serveur de fichiers à l'aide de classes d'exit», à la page 230.	Aucune valeur par défaut
protocolBridgeDataTimeout	Délai, en millisecondes, avant que l'agent de pont de protocole n'établisse une connexion de données avec un serveur FTP ou ne reçoive des données d'un serveur FTP par l'intermédiaire d'une connexion déjà établie. Si vous définissez cette propriété sur la valeur 0, l'agent de pont de protocole attend indéfiniment. Une fois le délai écoulé, l'agent de pont de protocole ferme toutes les connexions de données existantes au serveur FTP et tente d'établir une nouvelle connexion de données avant de reprendre le transfert en cours. Si la tentative visant à établir la nouvelle connexion de données échoue, le transfert en cours échoue également.	0
protocolBridgeLogoutBeforeDisconnect	Indique si l'agent de pont de protocole déconnecte l'utilisateur du serveur de fichiers avant de fermer la session FTP et de se déconnecter. Si vous définissez cette propriété sur true, l'agent de pont de protocole envoie une commande FTP QUIT au serveur de fichiers.	false
protocolBridgePropertiesConfiguration	Transmise comme l'une des propriétés de pont à la méthode initialize() des classes d'exit spécifiées par la propriété protocolBridgeServerPropertiesExitClasses.	Aucune valeur par défaut
protocolBridgePropertiesExitClasses	Spécifie une liste de classes implémentant une routine d'exit utilisateur des propriétés serveur du pont de protocole. Pour plus d'informations, voir la section «ProtocolBridgePropertiesExit2: recherche des propriétés du serveur de fichiers de protocole», à la page 224.	Aucune valeur par défaut

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
Propriétés de consignation de l'agent de pont de protocole :		
> V 9.0.0 > V 9.0.0 agentLog	Composant de la paire clé/valeur et opération permettant d'activer ou de désactiver la consignation des commandes FTP et des réponses entre l'agent de pont de protocole et les serveurs de fichiers FTP/SFTP/FTPS. Exemple : agentLog=on Active la consignation pour tous les composants agentLog=off Désactive la consignation pour tous les composants agentLog=ftp=on, sftp=on, ftps=off Active la consignation pour FTP et SFTP et la désactive pour FTPS	Aucune valeur par défaut
> V 9.0.0 > V 9.0.0 agentLogFileSize	Définit la taille maximale d'un fichier de capture en mégaoctets. Identique à la valeur par défaut de la taille de fichier standard par défaut de la trace.	20
> V 9.0.0 > V 9.0.0 agentLogFiles	Définit le nombre maximal de fichiers de capture conservés avant que le fichier le plus ancien ne soit supprimé.	10
> V 9.0.0 > V 9.0.0 agentLogFilter	Par défaut, capture les communications avec tous les serveurs FTP auxquels l'agent se connecte. Exemple : <ul style="list-style-type: none"> Filtrage sur le nom d'hôte/l'adresse IP <pre>host=ftpprod.ibm.com, ftp2.ibm.com host=9.182.*</pre> Filtrage basé sur les métadonnées <pre>metadata="outbound files to xyz corp"</pre> 	*
Propriétés de file d'attente :		
publicationMDUser	ID utilisateur MQMD à associer aux messages envoyés pour être publiés par le gestionnaire de file d'attente de coordination. Si vous ne définissez pas cette propriété, l'ID utilisateur MQMD est défini en fonction des règles IBM MQ de définition des ID utilisateur MQMD.	Aucune valeur par défaut
Propriétés de surveillance des ressources :		
monitorFilepathPlatformSeparator	Indique si des séparateurs de chemin spécifiques à la plateforme doivent être utilisés dans la variable \$FILEPATH. La valeur true utilise des séparateurs de chemin spécifiques à la plateforme. La valeur false utilise un séparateur de chemin de style UNIX (/) sur toutes les plateformes.	conforme
monitorMaxResourcesInPoll	Indique le nombre maximal de ressources surveillées à déclencher lors de chaque intervalle d'interrogation. Par exemple, si vous spécifiez le modèle de surveillance *.txt et un intervalle d'interrogation de 10 secondes et que vous définissez la propriété monitorMaxResourcesInPoll sur 10, la propriété monitorMaxResourcesInPoll limite les déclenchements de l'agent à 10 correspondances par intervalle d'interrogation. Les ressources correspondantes au delà de la limite de 10 sont déclenchées dans les intervalles d'interrogation ultérieurs. En outre, vous pouvez utiliser la propriété monitorMaxResourcesInPoll en combinaison avec un paramètre -bs correspondant sur la commande fteCreateMonitor , par exemple, pour restreindre chaque intervalle d'interrogation au déclenchement d'un seul transfert. Si vous spécifiez une valeur inférieure ou égale à zéro, le nombre de ressources de surveillance déclenchées dans un intervalle d'interrogation est illimité.	-1
monitorReportTriggerFail	Indique si les conditions d'échec, dans l'environnement et la configuration, détectées dans le moniteur, sont envoyées sous forme de message de journal à la rubrique SYSTEM.FTE. La valeur true consigne les messages. La valeur false ne consigne pas les messages.	conforme

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
monitorReportTriggerNotSatisfied	Indique si un déclencheur non satisfait envoie un message de journal à la rubrique SYSTEM.FTE qui contient les détails. La valeur <code>true</code> consigne les messages. La valeur <code>false</code> ne consigne pas les messages.	false
monitorReportTriggerSatisfied	Indique si un déclencheur satisfait envoie un message de journal à la rubrique SYSTEM.FTE qui contient les détails. La valeur <code>true</code> consigne les messages. La valeur <code>false</code> ne consigne pas les messages.	false
monitorSilenceOnTriggerFailure	Le nombre d'échecs consécutifs du déclencheur du moniteur de ressources avant les échecs n'est plus signalé.	5
monitorStopOnInternalFailure	Nombre de conditions FFDC internes consécutives du moniteur de ressources avant que le moniteur ne passe à l'état d'arrêt.	10
Propriétés du répertoire racine :		
commandPath	<p>Spécifie l'ensemble des chemins d'accès par lesquels les commandes peuvent être appelées, à l'aide de l'une des méthodes suivantes :</p> <ul style="list-style-type: none"> Tâches <code>Ant call</code>, <code>filecopy</code> ou <code>filemove</code> de l'agent Dans un message XML transmis à un agent, un des schémas XML de commande Managed File Transfer Agent pris en charge (par exemple, <code>managedCall</code> ou <code>managedTransfer</code>). <p>Pour plus d'informations sur la syntaxe valide de la valeur de la propriété <code>commandPath</code>, voir «Propriété <code>commandPath</code> MFT», à la page 374.</p> <p>Important : Soyez extrêmement prudent lorsque vous définissez cette propriété car toute commande de l'un des emplacements <code>commandPaths</code> spécifiés peut être effectivement appelée à partir d'un système client distant qui peut envoyer des commandes à l'agent. Pour cette raison, par défaut, lorsque vous spécifiez un emplacement <code>commandPath</code> :</p> <ul style="list-style-type: none"> Les bacs à sable d'agent existants sont configurés par l'agent au démarrage de sorte que tous les répertoires <code>commandPath</code> soient automatiquement ajoutés à la liste des répertoires auxquels l'accès à un transfert est refusé. Tous les bacs à sable d'utilisateur existants sont mis à jour au démarrage de l'agent de sorte que tous les répertoires <code>commandPath</code> (et leurs sous-répertoires) soient ajoutés en tant qu'éléments <code><exclude></code> aux éléments <code><read></code> et <code><write></code>. Si l'agent n'est pas configuré pour utiliser un bac à sable d'agent ou des bacs à sable d'utilisateur, un bac à sable d'agent est créé au démarrage de l'agent et les répertoires <code>commandPath</code> sont spécifiés comme répertoires refusés. <p>La valeur de cette propriété peut contenir des variables d'environnement.</p> <p>Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.</p> <p>V9.0.0.1 Vous pouvez définir la propriété <code>addCommandPathToSandbox</code> sur <code>false</code> pour remplacer ce comportement par défaut à des fins de compatibilité avec les versions suivantes :</p> <ul style="list-style-type: none"> IBM WebSphere MQ File Transfer Edition. Le composant IBM WebSphere MQ 7.5.0 Fix Pack 1 Managed File Transfer (ou version antérieure). Le composant IBM WebSphere MQ 7.5.0 Fix Pack 2 Managed File Transfer (ou version ultérieure) sur une installation pour laquelle la propriété d'installation <code>enableFunctionalFixPack=7502</code> n'est pas définie. <p>Important : Sachez que ce remplacement permet à un client de transférer toute commande vers le système de l'agent et d'appeler la commande et doit donc être utilisé avec précaution.</p>	Aucune - Aucune commande ne peut être appelée

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
<p>V9.0.0.1</p> <p>V9.0.0.1</p> <p>addCommandPathToSandbox</p>	<p>Indique si les répertoires spécifiés par la propriété <code>commandPath</code> (et tous leurs sous-répertoires) doivent être ajoutés :</p> <ul style="list-style-type: none"> Aux répertoires refusés d'un bac à sable d'agent existant. Les éléments <code><exclude></code> pour les éléments <code><read></code> et <code><write></code> pour les bacs à sable utilisateur qui ont été définis. A un nouveau bac à sable d'agent, si aucun agent n'a été configuré avec un bac à sable d'agent ou un ou plusieurs bacs à sable d'utilisateur. <p>Cela permet la compatibilité avec les versions suivantes :</p> <ul style="list-style-type: none"> IBM WebSphere MQ File Transfer Edition. Le composant IBM WebSphere MQ 7.5.0 Fix Pack 1 Managed File Transfer (ou version antérieure). Le composant IBM WebSphere MQ 7.5.0 Fix Pack 2 Managed File Transfer (ou version ultérieure) sur une installation pour laquelle la propriété <code>d'installation enableFunctionalFixPack=7502</code> n'est pas définie. <p>Pour plus d'informations, voir la section «Propriété <code>commandPath</code> MFT», à la page 374.</p>	True
<p>V9.0.1</p> <p>V9.0.0.1</p> <p>V9.0.1</p> <p>V9.0.0.1</p> <p>additionalWildcardSandboxChecking</p>	<p>Indique si des vérifications supplémentaires doivent être effectuées sur les transferts de caractères génériques pour un agent qui a été configuré avec un bac à sable d'utilisateur ou d'agent afin de restreindre les emplacements vers et depuis lesquels l'agent peut transférer des fichiers.</p> <p>Lorsque cette propriété est définie sur <code>true</code>, la vérification supplémentaire est activée. Si une demande de transfert tente de lire un emplacement qui se trouve en dehors du bac à sable défini pour la correspondance de fichier du caractère générique, le transfert échoue. S'il existe plusieurs transferts dans une même demande de transfert et que l'une de ces demandes échoue en raison d'une tentative de lecture d'un emplacement en dehors du bac à sable, le transfert complet échoue. Si la vérification échoue, la cause de l'échec est indiquée dans un message d'erreur (voir «Vérifications supplémentaires pour les transferts de caractères génériques», à la page 51).</p> <p>Si la propriété est omise ou définie sur <code>false</code>, aucune vérification supplémentaire n'est effectuée sur les transferts de caractères génériques.</p>	Aucun
sandboxRoot	<p>Spécifie l'ensemble des chemins d'accès racine à inclure et exclure lorsque vous utilisez les bacs à sable. Pour plus d'informations sur cette fonction, voir «Utilisation des bacs à sable d'agent MFT», à la page 46 .</p> <p>Séparez les chemins d'accès par des séparateurs de chemin d'accès spécifiques à la plateforme. Préfixez les chemins avec un point d'exclamation (!) pour indiquer que les chemins sont exclus du bac à sable. Cette fonction est utile si vous souhaitez exclure un sous-répertoire sous un chemin d'accès racine inclus.</p> <p>La propriété <code>sandboxRoot</code> n'est pas prise en charge sur les agents de pont de protocole.</p> <p>Vous ne pouvez pas spécifier les propriétés <code>sandboxRoot</code> et <code>userSandboxes</code> ensemble.</p> <p>La valeur de cette propriété peut contenir des variables d'environnement.</p> <p>Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.</p>	Aucune - Pas de bac à sable
transferRoot	<p>Répertoire racine par défaut pour les chemins d'accès relatifs indiqués à l'agent.</p> <p>La valeur de cette propriété peut contenir des variables d'environnement.</p> <p>Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.</p>	Répertoire de base de l'utilisateur ayant démarré le processus de l'agent.
transferRootHLQ	Qualificatif de haut niveau (ID utilisateur) par défaut pour les fichiers non complets spécifiés à l'agent	Nom d'utilisateur de l'utilisateur ayant démarré le processus de l'agent.
userSandboxes	<p>Restreint la zone du système de fichiers vers et depuis laquelle les fichiers peuvent être transférés en fonction du nom utilisateur MQMD de l'utilisateur qui demande le transfert. Pour plus d'informations, voir la section «Utilisation des bacs à sable utilisateur MFT», à la page 48.</p> <p>La propriété <code>userSandboxes</code> n'est pas prise en charge sur les agents de pont de protocole.</p> <p>Vous ne pouvez pas spécifier les propriétés <code>sandboxRoot</code> et <code>userSandboxes</code> ensemble.</p>	false
Propriété de planificateur :		

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
maxSchedulerRunDelay	Délai d'attente maximal, en minutes, avant que l'agent ne recherche les transferts planifiés. Spécifiez un entier positif pour activer cette propriété. Pour plus d'informations sur l'intérêt d'utiliser cette propriété, voir «Que faire si votre transfert de fichiers planifié ne s'exécute pas ou est retardé», à la page 321. L'agent pouvant lire une commande de sa file d'attente de commandes au moment où les transferts planifiés doivent être exécutés, un délai supplémentaire peut s'écouler avant que ces transferts planifiés ne démarrent. Dans ce cas, le planificateur est exécuté dès que cette commande est terminée.	-1
Propriétés de sécurité :		
agentQMGrAuthenticationCredentials File	Chemin d'accès au fichier qui contient les données d'identification de la connexion MQ.	 Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir Création d'un fichier de données d'identification MFT  Pour plus d'informations sur l'emplacement et les droits d'accès de ce fichier, voir Configuration de MQMFTCredentials.xml .  Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir MFT et IBM MQ authentification de la connexion .
authorityChecking	Indique si les fonctions de sécurité décrites dans «Restriction des droits utilisateur sur les actions de l'agent MFT», à la page 368 sont activées.   Depuis IBM MQ 9.0.0 Fix Pack 3 et IBM MQ 9.0.4, inquire est un droit requis sur toutes les files d'attente de droits d'accès de l'agent.	false
logAuthorityChecks	Niveau de consignation de la vérification des droits enregistré dans le journal des événements d'agent, dans le fichier output0.log. Les valeurs possibles pour cette propriété sont None, Failures et All.	Aucun
userIdForClientConnect	ID utilisateur transmis par l'intermédiaire des connexions client à IBM MQ. Si <i>java</i> est spécifié, le nom d'utilisateur signalé par la machine virtuelle Java est transmise dans le cadre de la demande de connexion à IBM MQ. Les valeurs possibles pour cette propriété sont None et <i>java</i> .	Aucun
Propriétés de délai d'expiration :		
maxTransferNegotiationTime	Délai maximal pendant lequel un transfert attend qu'un agent de destination termine la négociation. Si la négociation ne se termine pas dans le délai imparti, le transfert est placé dans un état de resynchronisation, ce qui permet l'exécution d'un autre transfert, quand ce dernier est disponible. Dans les scénarios où l'agent source ou de destination est soumis à une charge élevée, il se peut que la valeur par défaut soit trop faible pour que l'agent réponde assez rapidement à la demande de négociation. Ce cas se produit généralement lorsqu'un nombre important de moniteurs de ressources sont définis sur un agent source ou lorsque les moniteurs de ressources de cet agent surveillent des répertoires qui contiennent un grand nombre de fichiers. Il peut toutefois se produire quand un nombre important de demandes de transfert sont soumises à un agent. Il peut alors s'avérer nécessaire de spécifier une valeur supérieure ou égale à 200 000 pour cette propriété.	30 000
recoverableTransferRetryInterval	Délai d'attente en millisecondes entre la détection d'une erreur de transfert récupérable et une tentative de reprise du transfert.	60 000
senderTransferRetryInterval	Délai d'attente en millisecondes avant qu'un transfert rejeté soit relancé car la destination exécute déjà le nombre maximal de transferts. La valeur minimale est de 1 000.	30 000

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
transferAckTimeout	Délai en millisecondes pendant lequel un transfert attend l'accusé de réception ou des données de l'autre extrémité avant d'effectuer une nouvelle tentative. Il s'agit d'une propriété avancée et pour la plupart des configurations Managed File Transfer, il n'est pas nécessaire d'en modifier la valeur. Les accusés de réception sont envoyés par l'agent réceptionnaire à l'agent émetteur chaque fois qu'une fenêtre complète de données est reçue. Pour les réseaux non fiables ou dont la bande passante est limitée et les paramètres agentWindowSize et agentChunkSize dont la valeur est élevée, il est possible que la valeur par défaut ne suffise pas. Cela peut entraîner de nouveaux transferts inutiles entre les agents. Par conséquent, l'augmentation de cette valeur peut être bénéfique et permettre de réduire la probabilité du passage d'un transfert en mode reprise en raison d'un réseau lent.	60 000
transferAckTimeoutRetries	Nombre maximal de nouvelles tentatives d'accusé de réception pour un transfert sans obtenir de réponse avant que l'agent n'abandonne et ne place le transfert à l'état de récupération	5
xmlConfigReloadInterval	Intervalle en secondes entre les rechargements par l'agent des fichiers de configuration XML lors de la phase d'exécution. Pour empêcher l'agent de recharger les fichiers de configuration XML pendant la phase d'exécution, définissez cette propriété sur -1. Les fichiers de configuration XML suivants sont affectés par cette propriété : <ul style="list-style-type: none"> ConnectDirectCredentials.xml ConnectDirectNodeProperties.xml ConnectDirectProcessDefinitions.xml ProtocolBridgeCredentials.xml ProtocolBridgeProperties.xml UserSandboxes.xml 	30
Propriétés de trace et de consignation :		
javaCoreTriggerFile	Chemin d'accès complet à un emplacement de fichier surveillé par l'agent. Si le fichier existe à l'emplacement indiqué, le démarrage de l'agent déclenche un Javacore. Après avoir démarré l'agent, si vous mettez à jour un fichier à cet emplacement, l'agent déclenche à nouveau un fichier Javacore. Une unité d'exécution distincte interroge ce fichier toutes les 30 secondes pour déterminer si le fichier a été créé ou mis à jour. Si le fichier a été créé ou mis à jour depuis la dernière interrogation, l'agent génère un fichier Javacore dans l'un des répertoires suivants : <ul style="list-style-type: none"> UNIX: <code>MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name</code> Linux: <code>MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name</code> Windows: <code>MQ_DATA_PATH\mqft\logs\coordination_qmgr_name\agents\agent_name</code> Si vous spécifiez cette propriété, l'agent génère le message suivant au démarrage : <pre>BFGAG0092I The <insert_0> file will be used to request JVM diagnostic information.</pre> La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.	Aucun
trace	Spécification de trace si l'agent doit être exécuté avec la trace activée au démarrage de l'agent. La spécification de trace est une liste de classes séparée par des virgules, suivie du signe égal et d'un niveau de trace. Par exemple, <code>com.ibm.wmqfte.agent.Agent,com.ibm.wmqfte.commandhandler=all</code> . Vous pouvez spécifier plusieurs spécifications de trace dans une liste de valeurs séparées par deux-points. Par exemple, <code>com.ibm.wmqfte.agent.Agent=all:com.ibm.wmqfte.commandhandler=moderate</code> .	Aucun
outputLogFiles	Nombre total de fichiers output.log à conserver. Cette valeur s'applique au contrôleur de processus d'un agent et à l'agent lui-même.	5

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
outputLogSize	Taille maximale en Mo de chaque fichier output . log avant que la sortie ne passe au prochain fichier. Cette valeur s'applique au contrôleur de processus d'un agent et à l'agent lui-même.	1
outputLogEncoding	Codage de caractères utilisé par l'agent lorsqu'il écrit dans le fichier output . log.	Codage de caractères par défaut de la plateforme sur laquelle l'agent est exécuté.
traceFiles	Nombre total de fichiers de trace à conserver. Cette valeur s'applique au contrôleur de processus d'un agent, ainsi qu'à l'agent lui-même.	5
traceSize	Taille maximale en Mo de chaque fichier de trace avant que la trace ne passe au prochain fichier. Cette valeur s'applique au contrôleur de processus d'un agent et à l'agent lui-même.	20
traceMaxBytes	Quantité maximale de données de message générée dans le fichier de trace.	4096 octets
logTransferRecovery	Lorsque cette propriété est définie sur la valeur true, chaque fois qu'un transfert entre des événements de diagnostic de reprise est signalé dans le journal des événements de l'agent dans le fichier output0 . log .	Pour IBM MQ 9.0.0.0, la valeur par défaut est true
logCapture	Capture les messages de demande de transfert soumis à cet agent et les messages de journal publiés par l'agent sur le gestionnaire de file d'attente de coordination. Ces messages capturés peuvent s'avérer utiles lors du débogage des problèmes de transfert. Les messages capturés sont stockés dans des fichiers, dans le répertoire de journaux d'agent appelé capture? . log . ? est une valeur numérique. Le fichier qui contient le chiffre 0 contient les messages capturés les plus récents.	false
logCaptureFileSize	Définit la taille maximale d'un fichier de capture en mégaoctets.	10
logCaptureFiles	Définit le nombre maximal de fichiers de capture conservés avant que le fichier le plus ancien ne soit supprimé.	10
logCaptureFilter	Expression régulière Java utilisée par l'agent pour correspondre au nom de rubrique du message. Seuls les messages correspondant à l'expression régulière sont capturés.	.* (correspondance totale)
  Journal resourceMonitor	Paire clé-valeur du moniteur de ressources et de l'opération pour activer ou désactiver la journalisation. Les valeurs possibles sont les suivantes: <ul style="list-style-type: none"> • info • modéré • prolixe • désactivé Exemple : <ul style="list-style-type: none"> • resourceMonitorLog=MON1,MON2=info:MON3=off Activez la journalisation pour MON1 et MON2 et désactivez la journalisation pour MON3. • resourceMonitorLog=info Activez la journalisation de niveau info pour tous les moniteurs de ressources. Les journaux du moniteur de ressources sont écrits dans un fichier nommé resmoneventN . log, où N représente un nombre ; par exemple, resmonevent0 . log.  Avertissement : Tous les moniteurs de ressources d'un agent écrivent dans le même fichier journal. Pour plus d'informations, voir Journalisation des moniteurs de ressources MFT .	info
  Taille de resourceMonitorLogFile	Définit la taille maximale d'un fichier de capture en mégaoctets.	20
  resourceMonitorLogFiles	Définit le nombre maximal de fichiers de capture conservés avant que le fichier le plus ancien ne soit supprimé.	10
Propriétés de limite de transfert :		

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
maxDestinationTransfers	<p>Nombre maximal de transferts simultanés traités par l'agent de destination à tout moment. Chaque demande de transfert soumise à un agent est comptabilisée dans ce total, quel que soit le nombre de fichiers transférés pour satisfaire la demande. Cela signifie qu'une demande de transfert qui transfère un fichier unique compte autant qu'une demande de transfert qui transfère 10 fichiers.</p> <p>L'agent place les transferts en file d'attente lorsque l'agent de destination a atteint la limite spécifiée par la propriété maxDestinationTransfers.</p> <p>Si la somme des valeurs des propriétés maxSourceTransfers, maxDestinationTransfers et maxQueuedTransfers dépasse la valeur du paramètre MAXDEPTH de la file d'attente du magasin d'état (SYSTEM.FTE.STATE.nom_agent), l'agent ne démarre pas.</p>	<p>25 (pour tous les agents à l'exception de Connect:Direct)</p> <p>5 (pour les agents de pont Connect:Direct)</p>
maxFilesForTransfer	<p>Nombre maximal d'éléments de transfert autorisés pour un transfert géré unique. Si un transfert géré contient plus d'éléments que la valeur de maxFilesForTransfer, le transfert géré échoue et aucun élément de transfert n'est traité.</p> <p>La définition de cette propriété vous empêche de transférer accidentellement un trop grand nombre de fichiers en raison d'une demande de transfert incorrecte, par exemple, si un utilisateur spécifie accidentellement le transfert du répertoire racine / sur un système UNIX .</p>	5 000
maxSourceTransfers	<p>Nombre maximal de transferts simultanés traités par l'agent source à tout moment. Chaque demande de transfert soumise à un agent est comptabilisée dans ce total, quel que soit le nombre de fichiers transférés pour satisfaire la demande. Cela signifie qu'une demande de transfert qui transfère un fichier unique compte autant qu'une demande de transfert qui transfère 10 fichiers.</p> <p>L'agent source place les transferts en file d'attente lorsque l'agent de destination a atteint la limite spécifiée par la propriété maxSourceTransfers.</p> <p>Si la somme des valeurs des propriétés maxSourceTransfers, maxDestinationTransfers et maxQueuedTransfers dépasse la valeur du paramètre MAXDEPTH de la file d'attente du magasin d'état (SYSTEM.FTE.STATE.nom_agent), l'agent ne démarre pas.</p>	<p>25 (pour tous les agents, excepté les agents de pont Connect:Direct)</p> <p>5 (pour les agents de pont Connect:Direct)</p>
maxQueuedTransfers	<p>Nombre maximal de transferts en attente pouvant être mis en file d'attente par un agent source jusqu'à ce que l'agent rejette une nouvelle demande de transfert. Vous pouvez définir cette propriété de sorte que même si les limites des propriétés maxDestinationTransfers et maxSourceTransfers sont atteintes ou dépassées, les nouvelles demandes de transfert que vous effectuez maintenant sont acceptées, mises en file d'attente, puis réalisées ultérieurement.</p> <p>L'ordre suivant lequel les demandes de transfert en file d'attente sont traitées dépend de leur priorité et du temps qu'elles sont restées en file d'attente. Les transferts en attente anciens et à priorité élevée sont sélectionnés en premier. Les transferts à priorité basse qui sont restés longtemps dans la file d'attente sont sélectionnés de préférence par rapport aux transferts à priorité élevée plus récents.</p> <p>Si la somme des valeurs des propriétés maxSourceTransfers, maxDestinationTransfers et maxQueuedTransfers dépasse la valeur du paramètre MAXDEPTH de la file d'attente du magasin d'état (SYSTEM.FTE.STATE.nom_agent), l'agent ne démarre pas.</p>	1 000
<p>V 9.0.1 V 9.0.1 Propriétés de délai d'expiration pour la reprise des transferts :</p>		

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
<p>V 9.0.1</p> <p>V 9.0.1</p> <p>transferRecoveryTimeout</p>	<p>Délai, en secondes, au cours duquel un agent source continue de tenter de restaurer un transfert de fichier bloqué.</p> <p>Si cette propriété n'est pas définie, par défaut, l'agent effectue de nouvelles tentatives jusqu'à ce que la récupération du transfert aboutisse. Vous pouvez définir les valeurs suivantes pour la propriété de délai de reprise des transferts :</p> <p>-1 L'agent continue de tenter de reprendre le transfert bloqué jusqu'à ce que ce dernier soit terminé. Cette option est équivalente au comportement par défaut de l'agent si la propriété n'est pas définie.</p> <p>0 L'agent arrête le transfert de fichiers dès qu'il entre dans la phase de reprise.</p> <p>>0 L'agent continue de tenter de reprendre le transfert bloqué pendant le nombre de secondes défini par l'entier positif spécifié. Par exemple, <code>transferRecoveryTimeout=21600</code> indique que l'agent continue de tenter de reprendre le transfert pendant six heures à partir du moment où ce dernier est entré en phase de récupération. La valeur maximale de ce paramètre est 999999999.</p>	-1
Propriétés de routine d'exit utilisateur :		
agentForceConsistentPathDelimiters	Forcé le délimiteur de chemin dans les informations de fichier source et de fichier de destination fournies aux exits de transfert pour qu'il soit de type UNIX : barre oblique (/). Les options valides sont <code>true</code> et <code>false</code> .	false
destinationTransferEndExitClasses	Indique une liste de classes séparées par des virgules qui implémentent une routine d'exit utilisateur de transfert de destination.	Aucune valeur par défaut
destinationTransferStartExitClasses	Spécifie une liste de classes implémentant une routine d'exit utilisateur de début de transfert de destination.	Aucune valeur par défaut
exitClassPath	Indique une liste de répertoires, délimités par des caractères, spécifique à la plateforme, qui servent de chemins d'accès aux classes des routines d'exit utilisateur. Le répertoire des exits de l'agent est recherché avant toutes les entrées de ce chemin d'accès aux classes.	Répertoire exits de l'agent
exitNativeLibraryPath	Indique une liste de répertoires, délimités par des caractères, spécifique à la plateforme, qui servent de chemins d'accès aux bibliothèques natives des routines d'exit utilisateur.	Répertoire exits de l'agent
ioMaxRecordLength	Longueur maximale d'un enregistrement, en octets, pouvant être prise en charge pour un fichier orienté enregistrements. Managed File Transfer peut prendre en charge l'écriture dans des fichiers orientés enregistrement avec toute longueur d'enregistrement. Toutefois, des longueurs d'enregistrement importantes peuvent entraîner des erreurs de mémoire insuffisante. Par conséquent, pour éviter ces erreurs, la longueur d'enregistrement maximale est limitée par défaut à 64 ko. Lors de la lecture à partir de fichiers orientés enregistrement, un enregistrement entier doit tenir dans un seul bloc de transfert, de sorte que la longueur de l'enregistrement est en outre limitée par la taille du bloc de transfert. Cette propriété n'est utilisée que pour les fichiers orientés enregistrement des existes utilisateur d'E-S.	64 ko
monitorExitClasses	Spécifie une liste de classes implémentant une routine d'exit de moniteur, séparées par des virgules. Pour plus d'informations, voir la section « Exits utilisateur du moniteur de ressources MFT », à la page 902.	Aucune valeur par défaut
protocolBridgeCredentialExitClasses	Spécifie une liste de classes implémentant une routine d'exit utilisateur des données d'identification du pont de protocole. Pour plus d'informations, voir la section « Mappage des données d'identification pour un serveur de fichiers à l'aide de classes d'exit », à la page 230.	Aucune valeur par défaut
sourceTransferEndExitClasses	Spécifie une liste de classes implémentant une routine d'exit de fin de transfert source.	Aucune valeur par défaut
sourceTransferStartExitClasses	Spécifie une liste de classes implémentant une routine d'exit de début de transfert source.	Aucune valeur par défaut
IOExitClasses	Spécifie une liste de classes implémentant une routine d'exit utilisateur d'E-S, séparées par des virgules. Ne répertorie que les classes qui implémentent l'interface IOExit et non celles qui implémentent d'autres interfaces d'exit utilisateur d'E-S, comme IOExitResourcePath et IOExitChannel. Pour plus d'informations, voir la section « Utilisation des exits utilisateur d'E-S de transfert MFT », à la page 267.	Aucune valeur par défaut

Tableau 45. Propriétés d'agent avancées (suite)

Nom de la propriété	Description	Valeur par défaut
Compression du client IBM MQ :		
agentDataCompression	<p>Cette propriété n'est prise en charge que pour les connexions client.</p> <p>Liste de types de compression, séparés par des virgules, pour le transfert de données de fichier à négocier avec le serveur IBM MQ distant. Pour plus d'informations sur ces types de compression, reportez-vous à la rubrique suivante : Liste de compression des données de message.</p> <p>Les valeurs sont vérifiées pour vérifier leur validité, puis transmises par ordre d'apparence en tant que propriétés au canal client de l'agent. Le client IBM MQ gère ensuite la négociation entre ce canal client et le canal du serveur distant pour rechercher le plus petit dénominateur commun correspondant entre les propriétés de compression sur les deux canaux. Si aucune correspondance n'est trouvée, MQCOMPRESS_NONE est toujours sélectionné.</p>	MQCOMPRESS_NONE
agentHeaderCompression	<p>Cette propriété n'est prise en charge que pour les connexions client.</p> <p>Liste de types de compression, séparés par des virgules, pour le transfert de données d'en-tête à négocier avec le serveur IBM MQ distant. Les valeurs admises sont MQCOMPRESS_NONE et MQCOMPRESS_SYSTEM. Pour plus d'informations sur ces types de compression, reportez-vous à la rubrique suivante : Liste de compression des données d'en-tête.</p> <p>Les valeurs sont vérifiées pour vérifier leur validité, puis transmises par ordre d'apparence en tant que propriétés au canal client de l'agent. Le client IBM MQ gère ensuite la négociation entre ce canal client et le canal du serveur distant pour rechercher le plus petit dénominateur commun correspondant entre les propriétés de compression sur les deux canaux. Si aucune correspondance n'est trouvée, MQCOMPRESS_NONE est toujours sélectionné.</p>	MQCOMPRESS_NONE
<div style="display: flex; align-items: center;"> <div style="margin-right: 10px;"> </div> <div style="margin-right: 10px;"> </div> <div> Spécifiques à z/OS : </div> </div>		
<div style="display: flex; align-items: center;"> <div style="margin-right: 10px;"> </div> <div> adminGroup </div> </div>	<p>Un groupe de gestionnaires de sécurité. Les membres de ce groupe peuvent :</p> <ul style="list-style-type: none"> Démarrer l'agent à l'aide de la commande fteStartAgent. Arrêter l'agent à l'aide de la commande fteStopAgent. Activer ou désactiver la trace de l'agent à l'aide de la commande fteSetAgentTraceLevel. Affichez les détails d'un agent local en exécutant la commande fteShowAgentDetails avec le paramètre -d spécifié. <p>Définissez un groupe de gestionnaires de sécurité, par exemple <i>MFTADMIN</i>, puis ajoutez l'ID utilisateur de la tâche démarrée et les ID TSO de l'administrateur à ce groupe. Editez le fichier de propriétés de l'agent et définissez la propriété adminGroup pour qu'elle soit le nom de ce groupe de gestionnaires de sécurité.</p> <pre>adminGroup=MFTADMIN</pre>	Aucun
<div style="display: flex; align-items: center;"> <div style="margin-right: 10px;"> </div> <div style="margin-right: 10px;"> </div> <div> bpxwdynAllocAdditionalOptions </div> </div>	<p>Managed File Transfer utilise l'interface textuelle BPXWDYN pour créer et ouvrir des fichiers z/OS. Si BPXWDYN est utilisé pour l'allocation de fichier, par défaut, Managed File Transfer s'assure, si possible, que l'unité de données est montée (non requis pour les fichiers sur disque, mais requis pour les fichiers sur bande). Les options pouvant ne pas être prises en charge pour certains environnements, utilisez cette propriété pour modifier ce comportement. En outre, lors du transfert vers un fichier, il est également possible de spécifier des options pour BPXWDYN sur la ligne de commande ; ces options complètent celles spécifiées par cette propriété.</p> <p>Certaines options BPXWDYN ne doivent pas être spécifiées lors de l'utilisation de la propriété bpxwdynAllocAdditionalOptions dans le fichier <code>agent.properties</code>. Pour obtenir la liste de ces propriétés, voir «Propriétés BPXWDYN que vous ne devez pas utiliser avec MFT», à la page 661.</p>	<p>La valeur par défaut se présente comme suit :</p> <ul style="list-style-type: none"> MOUNT pour z/OS V1R8 et versions ultérieures
<div style="display: flex; align-items: center;"> <div style="margin-right: 10px;"> </div> <div style="margin-right: 10px;"> </div> <div> armELEMTYPE </div> </div>	<p>Propriété facultative. Si l'agent est configuré pour être redémarré par ARM (Automatic Restart Manager), définissez cette propriété sur la valeur du paramètre ARM ELEMTYPE spécifiée dans la règle ARM associée. Pour un agent, définissez ELEMTYPE sur SYSBFGAG.</p>	Non définie
<div style="display: flex; align-items: center;"> <div style="margin-right: 10px;"> </div> <div style="margin-right: 10px;"> </div> <div> armELEMENT </div> </div>	<p>Propriété facultative. Si l'agent est configuré pour être redémarré par ARM (Automatic Restart Manager), définissez cette propriété sur la valeur du paramètre ARM ELEMENT spécifiée dans la règle ARM associée. Vous pouvez définir la valeur ELEMENT de sorte qu'elle corresponde au nom de l'agent.</p>	Non définie

Tableau 45. Propriétés d'agent avancées (suite)		
Nom de la propriété	Description	Valeur par défaut
Autres propriétés :		
  legacyXMLMessageMQMDFormat	<p>Les messages XML de Managed File Transfer générés par l'agent (par exemple, les messages de journal et les messages de progression du transfert), sont désormais envoyés à une file d'attente avec une zone vide au format MQMD. Les versions précédentes du produit définissaient la zone au format MQMD sur MQSTR (chaîne de message texte). La définition de cette propriété sur true permet d'envoyer les messages XML de Managed File Transfer générés par l'agent à une file d'attente avec une zone au format MQMD définie sur MQSTR.</p> <p>Remarque : les messages de réponse de l'agent aux commandes seront envoyés avec un format de message correspondant à la demande de commande correspondante.</p> <p>Si la zone au format MQMD est définie sur MQSTR, les messages XML des commandes Managed File Transfer risquent d'être corrompus si la conversion des données est activée pour des canaux du réseau MQ.</p>	false
 adjustScheduleTimeForDaylightSavin g	<p>Si votre entreprise exécute des transferts planifiés tous les jours, car le transfert planifié a été créé avec:</p> <ul style="list-style-type: none"> • Paramètre -oi défini sur les jours, et • Paramètre -tb défini sur la source <p>sur la commande <code>fteCreateTransfer</code> par exemple, si vous affectez à cette propriété la valeur <code>true</code>, l'heure de transfert planifiée est avancée d'une heure lorsque les horloges sont avancées d'une heure et d'une heure lorsque les horloges sont reculées d'une heure.</p> <p>Par exemple, si votre transfert planifié doit être exécuté à 1h00, lorsque les horloges sont avancées, le transfert est exécuté à 2h00 et lorsque les horloges sont reculées, le transfert est rétabli à 1h00.</p>	conforme

Concepts associés

«Options de configuration de MFT sur Multiplatforms», à la page 58

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.

«Option de délai d'attente pour les transferts de fichiers en cours de récupération», à la page 253

Vous pouvez définir la durée, en secondes, pendant laquelle un agent source tente de récupérer un transfert de fichiers bloqué. Si le transfert n'aboutit pas lorsque l'agent atteint le délai d'attente pour l'intervalle entre les nouvelles tentatives, le transfert échoue.

«MFT bacs à sable», à la page 46

Vous pouvez restreindre la zone du système de fichiers à laquelle l'agent peut accéder dans le cadre d'un transfert. La zone à laquelle l'agent est limité est appelée le bac à sable. Vous pouvez appliquer des restrictions à l'agent ou à l'utilisateur qui demande un transfert.

Référence associée

«Propriétés système Java pour MFT», à la page 621

Un certain nombre de propriétés de commande et d'agent Managed File Transfer doivent être définies en tant que propriétés système Java, car elles définissent la configuration de la fonction précoce qui ne peut pas utiliser le mécanisme des propriétés de commande ou d'agent.

«Propriétés SSL pour MFT», à la page 622

Utilisez SSL ou TLS avec IBM MQ et Managed File Transfer pour empêcher les connexions non autorisées entre les agents et les gestionnaires de files d'attente et pour chiffrer le trafic des messages entre les agents et les gestionnaires de files d'attente.

«Agent `fteCreate`(créer un agent MFT)», à la page 399

La commande **`fteCreateAgent`** crée un Managed File Transfer Agent et sa configuration associée.

«`fteCreateBridgeAgent` (créer et configurer un agent de pont de protocole MFT)», à la page 403

La commande **`fteCreateBridgeAgent`** crée un agent de pont de protocole Managed File Transfer et sa configuration associée. Créez un agent de pont de protocole pour chaque serveur de fichiers auquel vous souhaitez envoyer des fichiers et duquel vous souhaitez les recevoir.

«fteCreateCDAgent (créer un agent de pont Connect:Direct)», à la page 413

La commande fteCreateCDAgent crée un Managed File Transfer Agent et sa configuration associée à utiliser avec le pont Connect:Direct .

«Utilisation des variables d'environnement dans les propriétés MFT», à la page 547

Depuis la IBM WebSphere MQ 7.5, il est possible d'utiliser des variables d'environnement dans les propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction des changements d'environnement. Par exemple, l'utilisateur qui exécute le processus.

Configuration d'un agent MFT pour plusieurs canaux dans un cluster

Si vous souhaitez utiliser la prise en charge multicanal d' 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 97.

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 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. 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.

Concepts associés

«Configuration de MFT pour la première utilisation», à la page 97

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.

Référence associée

«Le fichier MFT agent.properties», à la page 564

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.

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 97.

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.

Concepts associés

«Configuration de MFT pour la première utilisation», à la page 97

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.

Référence associée

«Le fichier MFT agent.properties», à la page 564

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.

enableUserMetadataOptions: clés de métadonnées MFT définies par l'utilisateur prises en charge

Lorsque la propriété d'agent **enableUserMetadataOptions** est définie sur true, les clés de métadonnées définies par l'utilisateur suivantes sont prises en charge lorsqu'elles sont spécifiées dans une nouvelle demande de transfert.

Nom de la clé	Description	Valeur par défaut
com.ibm.wmqfte.insertRecordLineSeparator	<p>Pour les transferts de texte. Lorsque cette clé est définie sur true, indique que lors de la lecture de fichiers orientés enregistrement, tels que des ensembles de données z/OS, des séparateurs de ligne doivent être insérés entre les enregistrements.</p> <p>Lorsque cette clé est définie sur false, indique que lors de la lecture de fichiers orientés enregistrement, les séparateurs de ligne ne doivent pas être insérés entre les enregistrements.</p>	conforme
com.ibm.wmqfte.newRecordOnLineSeparator	<p>Pour les transferts de texte. Lorsque cette clé est définie sur true, indique que lors de l'écriture dans des fichiers orientés enregistrement, tels que des ensembles de données z/OS, les séparateurs de ligne indiquent un nouvel enregistrement et ne sont pas écrits dans le cadre des données.</p> <p>Lorsque cette clé est définie sur false, indique que lors de l'écriture dans des fichiers orientés enregistrement, les séparateurs de ligne doivent être traités comme n'importe quel autre caractère (c'est-à-dire, aucun saut d'enregistrement).</p>	conforme

Tableau 46. Clés de métadonnées (suite)

Nom de la clé	Description	Valeur par défaut
com.ibm.wmqfte.convertLineSeparators	<p>Pour les transferts de texte. Indique si les séquences de séparateurs de ligne CRLF et LF sont converties en séquence de séparateurs de ligne requise pour la destination. Cette conversion ne prend actuellement effet que dans les cas suivants:</p> <ol style="list-style-type: none"> 1. Si la clé de métadonnées définie par l'utilisateur com.ibm.wmqfte.newRecordOnLineSeparator est définie sur false et que le transfert est effectué dans un fichier orienté enregistrement. 2. Si la clé de métadonnées définie par l'utilisateur com.ibm.wmqfte.com.ibm.wmqfte.insertRecordLineSeparator est définie sur false et que le transfert est effectué à partir d'un fichier orienté enregistrement. 	conforme

Information associée

«Tableau 45», à la page 566

Paramètre fteCreateTransfer -md

Fichiers de configuration d'agent MFT supplémentaires

Outre le fichier `agent.properties`, l'agent Managed File Transfer peut disposer d'un certain nombre de fichiers de configuration XML dans son répertoire de configuration.

Fichiers de configuration

Les fichiers de configuration XML suivants peuvent être utilisés pour spécifier des informations supplémentaires utilisées par l'agent:

ProtocolBridgeCredentials.xml

Si votre agent est un agent de pont de protocole, vous pouvez utiliser ce fichier pour spécifier les données d'identification à utiliser pour vous connecter au serveur FTP ou SFTP auquel l'agent se connecte.

ProtocolBridgeProperties.xml

Si votre agent est un agent de pont de protocole, vous pouvez utiliser ce fichier pour définir les propriétés des serveurs de fichiers de protocole autres que ceux par défaut auxquels l'agent se connecte. La commande **fteCreateBridgeAgent** crée un serveur de fichiers de protocole par défaut dans ce fichier pour vous.

ConnectDirectCredentials.xml

Si votre agent est un agent de pont Connect:Direct, vous pouvez utiliser ce fichier pour spécifier les données d'identification à utiliser pour la connexion aux noeuds Connect:Direct impliqués dans un transfert.

ConnectDirectNodeProperties.xml

Si votre agent est un agent de pont Connect:Direct, vous pouvez utiliser ce fichier pour spécifier les informations de système d'exploitation sur les noeuds Connect:Direct impliqués dans un transfert.

ConnectDirectProcessDefinition.xml

Si votre agent est un agent de pont Connect:Direct, vous pouvez utiliser ce fichier pour spécifier les processus Connect:Direct définis par l'utilisateur à appeler dans le cadre d'un transfert de fichier.

UserSandboxes.xml

Vous pouvez utiliser ce fichier pour spécifier les zones du système de fichiers dans lesquelles l'agent peut effectuer des opérations de lecture ou d'écriture.

Mise à jour des fichiers de configuration

Contrairement au fichier `agent.properties`, vous pouvez mettre à jour les fichiers de configuration XML et demander à l'agent de prendre en compte les modifications sans avoir à redémarrer l'agent.

Lorsque vous soumettez un transfert, s'il s'est produit plus de 10 secondes depuis la dernière vérification du fichier de configuration XML par l'agent, ce dernier vérifie l'heure de la dernière modification du fichier de configuration XML. Si le fichier de configuration XML a été modifié depuis la dernière lecture du fichier par l'agent, ce dernier le lit à nouveau. Si le contenu du fichier est valide par rapport au schéma XML, l'agent met à jour ses informations. Si le contenu du fichier n'est pas valide, l'agent utilise les informations de la version précédente du fichier et écrit un message dans le fichier `outpu0.log`.

Format de fichier des données d'identification du pont de protocole

Le fichier `ProtocolBridgeCredentials.xml` du répertoire de configuration Managed File Transfer Agent définit les noms d'utilisateur et les données d'identification que l'agent de pont de protocole utilise pour s'autoriser avec le serveur de protocole.

Le fichier `ProtocolBridgeCredentials.xml` doit être conforme au schéma `ProtocolBridgeCredentials.xsd`. Le document de schéma `ProtocolBridgeCredentials.xsd` se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/schema` de l'installation MQMFT. Les utilisateurs sont responsables de la création manuelle du fichier `ProtocolBridgeCredentials.xml`, qui n'est plus créé par la commande **fteCreateBridgeAgent**. Des exemples de fichier sont disponibles dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples` de l'installation MQMFT.

IBM WebSphere MQ 7.5 a introduit un nouvel élément `<agent>` qui contient l'élément `<server>` ou `<serverHost>` pour l'agent nommé.

Le fichier `ProtocolBridgeCredentials.xml` est rechargé périodiquement par l'agent et toute modification valide apportée au fichier aura une incidence sur le comportement de l'agent. L'intervalle de rechargement par défaut est de 30 secondes. Cet intervalle peut être modifié en spécifiant la propriété d'agent `xmlConfigReloadInterval` dans le fichier `agent.properties`.

Schéma- 7.5 ou version ultérieure

Le schéma suivant décrit les éléments valides dans le fichier `ProtocolBridgeCredentials.xml` pour IBM MQ 8.

```
<schema targetNamespace="http://wmqfte.ibm.com/ProtocolBridgeCredentials" elementFormDefault="qualified"
  xmlns="https://www.w3.org/2001/XMLSchema" xmlns:tns="http://wmqfte.ibm.com/
ProtocolBridgeCredentials">
  <!--
  <?xml version="1.0" encoding="UTF-8"?>
  <tns:credentials xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeCredentials"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeCredentials
  ProtocolBridgeCredentials.xsd ">
    <tns:agent name="agent1">
      <tns:serverHost name="myserver">
        <tns:user name="fred" serverPassword="pwd" serverUserId="bill"/>
        <tns:user name="jane" serverUserId="june" hostKey="1F:2e:f3">
          <tns:privateKey associationName="test" keyPassword="pwd2">
            ... private key ...
          </tns:privateKey>
        </tns:user>
      </tns:serverHost>
    </tns:agent>

    <tns:agent name="agent2">
      <tns:server name="server*" pattern="wildcard">
        <tns:user name="fred" serverPassword="pwd" serverUserId="bill"/>
        <tns:user name="jane" serverUserId="june" hostKey="1F:2e:f3">
          <tns:privateKey associationName="test" keyPassword="pwd2">
            ... private key ...
          </tns:privateKey>
        </tns:user>
      </tns:server>
    </tns:agent>

    <tns:agent name="agent3">
      <tns:serverHost name="ftpsServer"
        keyStorePassword="keypass"
        trustStorePassword="trustpass">
        <tns:user name="fred" serverPassword="pwd" serverUserId="bill"/>
      </tns:serverHost>
    </tns:agent>
  </!--
</schema>
```

```

        </tns:serverHost>
    </tns:agent>

</tns:credentials>
-->
<element name="credentials" type="tns:credentialsType"/>
<complexType name="credentialsType">
    <sequence>
        <element name="agent" type="tns:agentType" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
</complexType>
<complexType name="agentType">
    <choice minOccurs="0" maxOccurs="1">
        <element name="serverHost" type="tns:serverHostType" minOccurs="0" maxOccurs="unbounded"/>
        <element name="server" type="tns:serverType" minOccurs="0" maxOccurs="unbounded"/>
    </choice>
    <attribute name="name" type="string" use="required"/>
</complexType>
<complexType name="serverHostType">
    <sequence>
        <element ref="tns:user" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
    <attribute name="name" type="string" use="required"/>
    <attribute name="keyStorePassword" type="string" use="optional"/>
    <attribute name="keyStorePasswordCipher" type="string" use="optional"/>
    <attribute name="trustStorePassword" type="string" use="optional"/>
    <attribute name="trustStorePasswordCipher" type="string" use="optional"/>
</complexType>
<complexType name="serverType">
    <sequence>
        <element ref="tns:user" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
    <attribute name="name" type="string" use="required"/>
    <attribute name="pattern" type="tns:patternType" use="optional"/>
    <attribute name="keyStorePassword" type="string" use="optional"/>
    <attribute name="keyStorePasswordCipher" type="string" use="optional"/>
    <attribute name="trustStorePassword" type="string" use="optional"/>
    <attribute name="trustStorePasswordCipher" type="string" use="optional"/>
</complexType>
<element name="user" type="tns:userType"/>
<complexType name="userType">
    <sequence>
        <element ref="tns:privateKey" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
    <attribute name="name" type="string" use="required"/>
    <attribute name="serverUserId" type="string" use="optional"/>
    <attribute name="serverUserIdCipher" type="string" use="optional"/>
    <attribute name="serverPassword" type="string" use="optional"/>
    <attribute name="serverPasswordCipher" type="string" use="optional"/>
    <attribute name="hostKey" use="optional">
        <simpleType>
            <restriction base="string">
                <pattern
                    value="([a-zA-F0-9]){2}(:([a-zA-F0-9]){2})*">
                </pattern>
            </restriction>
        </simpleType>
    </attribute>
</complexType>
<element name="privateKey" type="tns:privateKeyType"/>
<complexType name="privateKeyType">
    <simpleContent>
        <extension base="string">
            <attribute name="keyPassword" type="string" use="optional"/>
            <attribute name="keyPasswordCipher" type="string" use="optional"/>
            <attribute name="associationName" type="string" use="required"/>
        </extension>
    </simpleContent>
</complexType>
<!--
    Determines the type of pattern matching to use.
-->

```

```

<simpleType name="patternType">
  <restriction base="string">
    <enumeration value="regex"/>
    <enumeration value="wildcard"/>
  </restriction>
</simpleType>
</schema>

```

Description du fichier ProtocolBridgeCredentials.xml

Les éléments et attributs utilisés dans le fichier ProtocolBridgeCredentials.xml sont décrits dans la liste suivante.

< données_d'identification >

Élément de groupe contenant des éléments qui décrivent les données d'identification utilisées par un agent de pont de protocole pour se connecter à un serveur de protocole.

< agent >

Élément contenant une définition < server > ou < serverHost > pour un agent nommé.

< server >

Serveur de protocole auquel le pont de protocole se connecte.

L'élément < server > n'est pas pris en charge pour 7.0.4 ou version antérieure.

Attribut	Description
nom	Nom du serveur de protocole.
schéma	Si vous avez utilisé des caractères génériques ou des expressions régulières pour spécifier le modèle d'un nom de serveur de protocole, utilisez wildcard ou regex.
trustStoreMot de passe ou trustStorePasswordCipher	Obligatoire lorsque l'élément < serveur > fait référence à un serveur FTPS. Mot de passe utilisé pour accéder au magasin de clés de confiance. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
keyStoreMot de passe ou keyStorePasswordCipher	Facultatif. Mot de passe utilisé pour accéder au magasin de clés. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.

< serverHost >

Nom d'hôte du serveur de protocole auquel le pont de protocole se connecte.

Le fichier ProtocolBridgeCredentials.xml peut contenir des éléments < serverHost > ou des éléments < server >, mais vous ne pouvez pas utiliser une combinaison des deux types différents. Lorsque vous utilisez < serverHost >, le nom est comparé au nom d'hôte du serveur de protocole. Lorsque vous utilisez < server >, le nom est comparé au nom du serveur de protocole (tel que défini dans le fichier ProtocolBridgeProperties.xml).

Attribut	Description
nom	Nom d'hôte ou adresse IP du serveur de protocole.
trustStoreMot de passe ou trustStorePasswordCipher	Obligatoire lorsque l'élément < serverHost > fait référence à un serveur FTPS. Mot de passe utilisé pour accéder au magasin de clés de confiance. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
keyStoreMot de passe ou keyStorePasswordCipher	Facultatif. Mot de passe utilisé pour accéder au magasin de clés. Cette propriété est facultative sauf si vous définissez l'attribut

Attribut	Description
	keyStore , auquel cas elle est obligatoire. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.

< utilisateur >

Mappage d'un nom d'utilisateur Managed File Transfer à un nom d'utilisateur de serveur de protocole.

Attribut	Description
nom	Nom d'utilisateur utilisé avec Managed File Transfer.
serverUserID ou serverUserIdCipher	Nom d'utilisateur utilisé avec le serveur de protocole. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
Chiffrement serverPassword ou serverPassword	Mot de passe du nom d'utilisateur utilisé sur le serveur de protocole. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
hostKey	Empreinte digitale SSH de l'hôte du serveur.

<privateKey >

Clé privée d'un utilisateur.

Attribut	Description
keyPassword ou keyStorePasswordCipher	Mot de passe de la clé privée. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
associationName	Nom utilisé pour la trace et la consignation.

Format du fichier de propriétés du pont de protocole

Le fichier ProtocolBridgeProperties.xml du répertoire de configuration de l'agent définit les propriétés des serveurs de fichiers de protocole.

Le fichier ProtocolBridgeProperties.xml doit être conforme au schéma ProtocolBridgeProperties.xsd . Le document de schéma ProtocolBridgeProperties.xsd se trouve dans le répertoire MQ_INSTALLATION_PATH/mqft/samples/schema de l'installation Managed File Transfer . Un fichier modèle, ProtocolBridgeProperties.xml, est créé par la commande **fteCreateBridgeAgent** dans le répertoire de configuration de l'agent.

Le fichier ProtocolBridgeProperties.xml est rechargé périodiquement par l'agent et toute modification valide apportée au fichier aura une incidence sur le comportement de l'agent. L'intervalle de rechargement par défaut est de 30 secondes. Vous pouvez modifier cet intervalle en spécifiant la propriété d'agent xmlConfigReloadInterval dans le fichier agent.properties .

Schéma

Le schéma suivant décrit le fichier ProtocolBridgeProperties.xml .

Remarque : Les attributs maxReconnectRetry et reconnectWaitPeriod ne sont pas pris en charge sous IBM WebSphere MQ 7.5, IBM WebSphere MQ File Transfer Edition 7.0.2ou version ultérieure.

```
<schema targetNamespace="http://wmqfte.ibm.com/ProtocolBridgeProperties" elementFormDefault="qualified"
  xmlns="https://www.w3.org/2001/XMLSchema" xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeProperties">
  <!--
  Example: ProtocolBridgeProperties.xml

  <?xml version="1.0" encoding="UTF-8"?>
  <tns:serverProperties xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeProperties"
```

```

        xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://wmmqfte.ibm.com/ProtocolBridgeProperties
        ProtocolBridgeProperties.xsd">
<tns:credentialsFile path="$HOME/ProtocolBridgeCredentials.xml"/>
<tns:defaultServer name="myserver"/>
<tns:ftpServer name="myserver" host="myhost.hursley.ibm.com" port="1234" platform="windows"
        timeZone="Europe/London" locale="en-GB" fileEncoding="UTF-8"
        listFormat="unix" limitedWrite="false"/>
<tns:sftpServer name="server1" host="myhost.hursley.ibm.com" platform="windows"
        fileEncoding="UTF-8" limitedWrite="false">
    <limits maxListFileNames="10"/>
</tns:sftpServer>
</tns:serverProperties>
-->

<!-- Root element for the document -->
<element name="serverProperties" type="tns:serverPropertiesType"></element>

<!--
    A container for all protocol bridge server properties
-->
<complexType name="serverPropertiesType">
    <sequence>
        <element name="credentialsFile" type="tns:credentialsFileName" minOccurs="0" maxOccurs="1"/>
        <element name="defaultServer" type="tns:serverName" minOccurs="0" maxOccurs="1"/>
        <choice minOccurs="0" maxOccurs="unbounded">
            <element name="ftpServer" type="tns:ftpServerType"/>
            <element name="sftpServer" type="tns:sftpServerType"/>
            <element name="ftpsServer" type="tns:ftpsServerType"/>
            <element name="ftpsfgServer" type="tns:ftpsfgServerType"/>
            <element name="ftpsfgServer" type="tns:ftpsfgServerType"/>
        </choice>
    </sequence>
</complexType>

<!--
    A container for a server name
-->
<complexType name="serverName">
    <attribute name="name" type="tns:serverNameType" use="required"/>
</complexType>

<!--
    A container for a credentials file name
-->
<complexType name="credentialsFileName">
    <attribute name="path" type="string" use="required"/>
</complexType>

<!--
    A container for all the information about an FTP server
-->
<complexType name="ftpServerType">
    <sequence>
        <element name="limits" type="tns:generalLimitsType" minOccurs="0" maxOccurs="1" />
    </sequence>
    <attributeGroup ref="tns:ftpServerAttributes"/>
    <attribute name="passiveMode" type="boolean" use="optional"/>
</complexType>

<!--
    A container for all the information about an SFG FTP server
-->
<complexType name="ftpsfgServerType">
    <sequence>
        <element name="limits" type="tns:generalLimitsType" minOccurs="0" maxOccurs="1" />
    </sequence>
    <attributeGroup ref="tns:ftpServerAttributes"/>
</complexType>

<!--
    A container for all the information about an SFTP server
-->
<complexType name="sftpServerType">
    <sequence>
        <element name="limits" type="tns:sftpLimitsType" minOccurs="0" maxOccurs="1" />
    </sequence>
    <attributeGroup ref="tns:sftpServerAttributes"/>
</complexType>

<!--
    A container for all the information about a FTPS server
-->

```

```

-->
<complexType name="ftpsServerType">
  <sequence>
    <element name="limits" type="tns:generalLimitsType" minOccurs="0" maxOccurs="1" />
  </sequence>
  <attributeGroup ref="tns:ftpsServerAttributes"/>
</complexType>

<!--
  A container for all the information about a SFG FTPS server
-->
<complexType name="ftpsSfgServerType">
  <sequence>
    <element name="limits" type="tns:generalLimitsType" minOccurs="0" maxOccurs="1" />
  </sequence>
  <attributeGroup ref="tns:ftpsServerAttributes"/>
</complexType>

<!--
  Attributes common to all server types
-->
<attributeGroup name="generalServerAttributes">
  <attribute name="name" type="tns:serverNameType" use="required"/>
  <attribute name="host" type="string" use="required"/>
  <attribute name="port" type="nonNegativeInteger" use="optional"/>
  <attribute name="platform" type="tns:platformType" use="required"/>
  <attribute name="fileEncoding" type="string" use="required"/>
  <attribute name="limitedWrite" type="boolean" use="optional"/>
  <attribute name="controlEncoding" type="string" use="optional"/>
</attributeGroup>

<!--
  Attributes common to ftp and ftps server types
-->
<attributeGroup name="ftpServerAttributes">
  <attributeGroup ref="tns:generalServerAttributes"/>
  <attribute name="timeZone" type="string" use="required"/>
  <attribute name="locale" type="tns:localeType" use="required"/>
  <attribute name="listFormat" type="tns:listFormatType" use="optional"/>
  <attribute name="listFileRecentDateFormat" type="tns:dateFormatType" use="optional"/>
  <attribute name="listFileOldDateFormat" type="tns:dateFormatType" use="optional"/>
  <attribute name="monthShortNames" type="tns:monthShortNamesType" use="optional"/>
</attributeGroup>

<!--
  Attributes common to ftps server types
-->
<attributeGroup name="ftpsServerAttributes">
  <attributeGroup ref="tns:ftpServerAttributes"/>
  <attribute name="ftpsType" type="tns:ftpsTypeType" use="optional"/>
  <attribute name="trustStore" type="string" use="required"/>
  <attribute name="trustStoreType" type="string" use="optional"/>
  <attribute name="keyStore" type="string" use="optional"/>
  <attribute name="keyStoreType" type="string" use="optional"/>
  <attribute name="ccc" type="boolean" use="optional"/>
  <attribute name="protFirst" type="boolean" use="optional"/>
  <attribute name="auth" type="string" use="optional"/>
  <attribute name="connectTimeout" type="nonNegativeInteger" use="optional"/>
</attributeGroup>

<!--
  A container for limit-type attributes for a server. Limit parameters
  are optional, and if not specified a system default will be used.
-->
<complexType name="generalLimitsType">
  <attributeGroup ref="tns:generalLimitAttributes"/>
</complexType>

<complexType name="sftpLimitsType">
  <attributeGroup ref="tns:generalLimitAttributes"/>
  <attribute name="connectionTimeout" type="nonNegativeInteger" use="optional"/>
</complexType>

<!--
  Attributes for limits common to all server types
-->
<attributeGroup name="generalLimitAttributes">
  <attribute name="maxListFileNames" type="positiveInteger" use="optional"/>
  <attribute name="maxListDirectoryLevels" type="nonNegativeInteger" use="optional"/>
  <attribute name="maxReconnectRetry" type="nonNegativeInteger" use="optional"/>

```

```

    <attribute name="reconnectWaitPeriod" type="nonNegativeInteger" use="optional"/>
    <attribute name="maxSessions" type="positiveInteger" use="optional"/>
    <attribute name="socketTimeout" type="nonNegativeInteger" use="optional"/>
</attributeGroup>

<!--
and
    The type for matching valid server names. Server names must be at least 2 characters in length
    are limited to alphanumeric characters and the following characters: ".", "_", "/" and "%".
-->
<simpleType name="serverNameType">
    <restriction base="string">
        <pattern value="[0-9a-zA-Z\._%]{2,}" />
    </restriction>
</simpleType>

<!--
-->
    The types of platform supported.
-->
<simpleType name="platformType">
    <restriction base="string">
        </restriction>
</simpleType>

<!--
-->
    The type for matching a locale specification.
-->
<simpleType name="localeType">
    <restriction base="string">
        <pattern value="(.)[-_](.)" />
    </restriction>
</simpleType>

<!--
-->
    The types of list format supported (for FTP servers).
-->
<simpleType name="listFormatType">
    <restriction base="string">
        </restriction>
</simpleType>

<!--
-->
    Date format for FTP client directory listing on an FTP server. This is
    the format to be passed to methods setDefaultDateFormatStr and
    setRecentDateFormatStr for Java class:
    org.apache.commons.net.ftp.FTPClientConfig
-->
<simpleType name="dateFormatType">
    <restriction base="string">
        </restriction>
</simpleType>

<!--
-->
    A list of language-defined short month names can be specified. These are
    used for translating the directory listing received from the FTP server.
    The format is a string of three character month names separated by "|"
-->
<simpleType name="monthShortNamesType">
    <restriction base="string">
        <pattern value="(...\|){11}{...}" />
    </restriction>
</simpleType>

<!--
-->
    The enumerations of the allowed FTPS types: "implicit" & "explicit"
    If not specified the default is "explicit"
-->
<simpleType name="ftpsTypeType">
    <restriction base="string">
        <enumeration value="explicit" />
        <enumeration value="implicit" />
    </restriction>
</simpleType>

<!--
-->
    Attribute Group for SFTP Servers
-->
<attributeGroup name="sftpServerAttributes">
    <attributeGroup ref="tns:generalServerAttributes" />
    <attribute name="cipherList" type="string" use="optional" />

```

```
</attributeGroup>  
</schema>
```

Description du fichier ProtocolBridgeProperties.xml

Les éléments et attributs utilisés dans le fichier ProtocolBridgeProperties.xml sont décrits dans la liste suivante:

<serverProperties>

Élément racine du document XML

<credentialsFile>

Chemin d'accès au fichier contenant les données d'identification. Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir [«Utilisation des variables d'environnement dans les propriétés MFT»](#), à la page 547

<defaultServer>

Serveur de fichiers de protocole qui fait office de serveur par défaut pour les transferts de fichiers

<ftpServer>

Un serveur de fichiers FTP

<sftpServer>

Un serveur de fichiers SFTP

<ftpsServer>

Un serveur de fichiers FTPS

Attributs de serveur généraux qui s'appliquent à tous les types de serveur de fichiers de protocole:

Attribut	Description
nom	Obligatoire. Nom du serveur de fichiers de protocole. Les noms de serveur de protocole doivent comporter au moins deux caractères, ne sont pas sensibles à la casse et sont limités aux caractères alphanumériques et aux caractères suivants: <ul style="list-style-type: none">• Point (.)• trait de soulignement (_)• barre oblique (/)• signe pourcentage (%)
host	Obligatoire. Nom d'hôte ou adresse IP du serveur de fichiers de protocole à partir duquel vous souhaitez envoyer ou recevoir des fichiers.
port	Facultatif. Numéro de port du serveur de fichiers de protocole à partir duquel vous souhaitez envoyer ou recevoir des fichiers.
plateforme	Obligatoire. Plateforme du serveur de fichiers de protocole à partir de laquelle vous souhaitez envoyer ou recevoir des fichiers. Indiquez UNIX ou WINDOWS. Définissez cette propriété en fonction de la manière dont vous entrez les chemins d'accès sur votre serveur FTP, FTPS ou SFTP. Par exemple, si vous exécutez un serveur FTP sous Windows, mais que vous vous connectez au serveur, vous devez entrer des chemins de style UNIX (c'est-à-dire avec des barres obliques), définir cette valeur sur UNIX et non sur WINDOWS. Les serveurs s'exécutant sous Windows présentent souvent un système de fichiers de style UNIX.
fileEncoding	Obligatoire. Définit le codage de caractères utilisé par le serveur de fichiers. Cette propriété est utilisée lorsque vous transférez des fichiers en mode texte afin que les séquences de codage correctes soient modifiées lorsque les fichiers sont déplacés entre les plateformes. Par exemple, UTF-8.

Attribut	Description
limitedWrite	Facultatif. Le mode par défaut lors de l'écriture sur un serveur de fichiers consiste à créer un fichier temporaire, puis à renommer ce fichier lorsque le transfert est terminé. Pour un serveur de fichiers configuré en écriture seule, le fichier est créé directement avec son nom final. La valeur de cette propriété peut être true ou false. La valeur par défaut est false.
controlEncoding	Facultatif. Valeur de codage de contrôle pour les messages de contrôle envoyés au serveur de fichiers de protocole. Cette propriété affecte le codage du nom de fichier utilisé et doit être compatible avec le codage de contrôle du serveur de fichiers de protocole. Valeur par défaut : UTF-8.

Attributs généraux qui s'appliquent uniquement aux serveurs FTP et FTPS:

Attribut	Description
timeZone	Obligatoire. Fuseau horaire du serveur de fichiers de protocole à partir duquel vous souhaitez envoyer ou recevoir des fichiers. Par exemple: America / New_York ou Asia / Tokyo.
paramètres_régionaux	Obligatoire. Langue utilisée sur le serveur de fichiers de protocole pour l'envoi ou la réception de fichiers. Par exemple: en_US ou ja_JP
listFormat	Facultatif. Format de liste qui définit le format des informations de liste de fichiers renvoyées par le serveur de fichiers de protocole. Utilisez Windows ou UNIX. La valeur par défaut est UNIX.
Format listFileRecentDate	Facultatif. Format de date récent (moins d'un an) pour la liste des répertoires de client FTP sur un serveur FTP. Cet attribut et l'attribut de format listFileOldDate vous permettent de redéfinir les formats de date attendus renvoyés par le serveur de fichiers de protocole. La valeur par défaut est celle définie par le serveur de fichiers de protocole.
Format listFileOldDate	Facultatif. Ancien format de date (plus d'un an) pour la liste des répertoires de client FTP sur un serveur FTP. Cet attribut et l'attribut de format listFileRecentDate vous permettent de redéfinir les formats de date attendus renvoyés par le serveur de fichiers de protocole. La valeur par défaut est celle définie par le serveur de fichiers de protocole.
Noms monthShort	Facultatif. Liste de remplacement des noms de mois utilisés pour décoder les informations de date renvoyées par le serveur de fichiers de protocole. Cette propriété est constituée d'une liste de 12 noms, séparés par des virgules, pour remplacer les valeurs de mois de l'environnement local par défaut. La valeur par défaut est celle définie par le serveur de fichiers de protocole.

Attributs généraux qui s'appliquent uniquement aux serveurs FTP:

Attribut	Description
passiveMode	Facultatif. Contrôle si la connexion au serveur FTP est passive ou active. Si vous définissez la valeur de cette propriété sur false, la connexion est active. Si vous définissez la valeur sur true, la connexion est passive. La valeur par défaut est false.

Attributs généraux qui s'appliquent uniquement aux serveurs FTPS:

Attribut	Description
ftpsType	Facultatif. Indique si la forme explicite ou implicite du protocole FTPS est utilisée. La valeur par défaut est <code>explicit</code> .
trustStore	Obligatoire. Emplacement du magasin de clés de confiance utilisé pour déterminer si le certificat présenté par le serveur FTPS est digne de confiance.
trustStoreType	Facultatif. Format du fichier de clés certifiées. La valeur par défaut est <code>JKS</code> .
keyStore	Facultatif. Emplacement du magasin de clés utilisé pour fournir des informations de certificat en cas d'authentification par le serveur FTPS. Par défaut, le pont de protocole ne peut pas se connecter aux serveurs FTPS configurés pour exiger l'authentification des clients.
KeyStoreType	Facultatif. Format du fichier de clés. La valeur par défaut est <code>JKS</code> .
CCC	Facultatif. Indique si un canal de commande de suppression (non chiffré) est utilisé lorsque l'authentification est terminée. La valeur par défaut est <code>false</code> , ce qui signifie que le canal de commande reste chiffré pendant toute la durée de la session FTPS. Cet attribut est applicable uniquement lorsque <code>ftpsType</code> est défini sur <code>explicit</code> .
protFirst	Facultatif. Indique si les commandes USER/PASS sont émises vers le serveur FTPS avant ou après les commandes PBSZ/PROT . La valeur par défaut est <code>false</code> , ce qui signifie que les commandes USER/PASS sont envoyées en premier, suivies des commandes PBSZ/PROT . Cet attribut est applicable uniquement lorsque <code>ftpsType</code> est défini sur <code>explicit</code> .
auth	Facultatif. Indique le protocole spécifié dans le cadre de la commande AUTH . Un protocole spécifié sera essayé en premier, puis la valeur par défaut est d'essayer TLS, SSL, TLS-Cou TLS-P jusqu'à ce que le serveur FTPS ne soit pas rejeté avec un code de réponse 504. Cet attribut est applicable uniquement lorsque <code>ftpsType</code> est défini sur <code>explicit</code> .

< limites >

Élément de conteneur pour les attributs communs à tous les types de serveur et pour les attributs spécifiques à un type de serveur:

Attributs de limite générale qui s'appliquent à tous les types de serveur de fichiers de protocole:

Attribut	Description
maxListFileNames	Facultatif. Nombre maximal de noms collectés lors de l'analyse d'un répertoire sur le serveur de fichiers de protocole à la recherche de noms de fichier. La valeur par défaut est <code>999999999</code> .
maxListDirectoryLevels	Facultatif. Nombre maximal de niveaux de répertoire sur le serveur de protocole pour l'analyse récursive des noms de fichier. La valeur par défaut est <code>1000</code> .
maxReconnectRelance (Cet attribut est désormais obsolète.)	Obsolète. Cet attribut n'est pas pris en charge sous IBM WebSphere MQ 7.5 ou sous IBM WebSphere MQ File Transfer Edition 7.0.2 ou version ultérieure. Facultatif. Nombre maximal de fois où un serveur de protocole tente de se reconnecter avant que l'agent de pont de protocole n'arrête d'essayer. La valeur par défaut est <code>2</code> .
Période reconnectWait	Obsolète. Cet attribut n'est pas pris en charge sous IBM WebSphere MQ 7.5 ou sous IBM WebSphere MQ File Transfer Edition 7.0.2 ou version ultérieure.

Attribut	Description
(Cet attribut est désormais obsolète.)	Facultatif. Délai d'attente, en secondes, avant la tentative de reconnexion. La valeur par défaut est de 10 secondes.
maxSessions	Facultatif. Nombre maximal de sessions pour le serveur de protocole. Ce nombre doit être supérieur ou égal à la somme du nombre maximal de transferts source et de destination pour l'agent de pont de protocole. La valeur par défaut correspond à la somme des valeurs des propriétés d'agent maxSourceTransferts, maxDestinationTransferts et maxCommandHandlerThreads, plus 1. Si ces trois propriétés utilisent leurs valeurs par défaut de 25, 25 et 5, la valeur par défaut de maxSessions est 56.
socketTimeout	Facultatif. Délai d'attente du socket en secondes. La valeur de cet attribut est utilisée lors de la diffusion en flux de fichiers. La valeur par défaut est fixée à 30 secondes.

Attribut de limite qui s'applique uniquement aux serveurs SFTP:

Attribut	Description
connectionTimeout	Facultatif. Délai d'attente, en secondes, d'une réponse du serveur de fichiers de protocole à une demande de connexion. Un délai d'expiration indique que le serveur de fichiers de protocole n'est pas disponible. La valeur par défaut est de 30 secondes.
cipherList	<p>Facultatif. Indique une liste de chiffrements séparés par des virgules qui sont utilisés pour communiquer entre l'agent de pont de protocole et le serveur SFTP. Les chiffrements sont appelés dans l'ordre dans lequel ils sont spécifiés dans cette liste. Le chiffrement doit être disponible sur le serveur et sur le client avant de pouvoir être utilisé.</p> <p>Les chiffrements pris en charge par l'agent de pont de protocole sont les suivants:</p> <ul style="list-style-type: none"> • fellation-cbc • 3des-cbc • aes128-cbc • aes192-cbc • aes256-cbc • aes128-ctr • aes192-ctr • aes256-ctr • 3des-ctr • arc quatre • arcfour128 • arcfour256 <p>Par défaut, la liste des chiffrements utilisés par les agents de pont de protocole est aes128-cbc , aes192-cbc , aes256-cbc.</p>

Format de fichier de données d'identification Connect:Direct

Le fichier ConnectDirectCredentials.xml du répertoire de configuration Managed File Transfer Agent définit les noms d'utilisateur et les données d'identification utilisés par l'agent Connect:Direct pour s'autoriser avec un noeud Connect:Direct .

Le fichier `ConnectDirectCredentials.xml` doit être conforme au schéma `ConnectDirectCredentials.xsd`. Le document de schéma `ConnectDirectCredentials.xsd` se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/schema` de l'installation MQMFT. Un exemple de fichier `ConnectDirectCredentials.xml` se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/credentials` de l'installation MQMFT.

Le fichier `ConnectDirectCredentials.xml` est rechargé périodiquement par l'agent et toute modification valide apportée au fichier aura une incidence sur le comportement de l'agent. L'intervalle de rechargement par défaut est de 30 secondes. Cet intervalle peut être modifié en spécifiant la propriété d'agent `xmlConfigReloadInterval` dans le fichier `agent.properties`.

Schéma

Le schéma suivant décrit les éléments valides dans le fichier `ConnectDirectCredentials.xml`.

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  This schema defines the format of the XML file that is located in the agent properties
  directory of a Connect:Direct bridge agent. The XML file ConnectDirectCredentials.xml
  is used by the default credential validation of the Connect:Direct bridge.
  For more information, see the WebSphere MQ InfoCenter
-->

<schema targetNamespace="http://wmqfte.ibm.com/ConnectDirectCredentials"
  elementFormDefault="qualified"
  xmlns="https://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://wmqfte.ibm.com/ConnectDirectCredentials"

  <!--
    <?xml version="1.0" encoding="UTF-8"?>

    <tns:credentials xmlns:tns="http://wmqfte.ibm.com/ConnectDirectCredentials"
      xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://wmqfte.ibm.com/ConnectDirectCredentials
        ConnectDirectCredentials.xsd">
      <tns:agent name="CDAGENT01">
        <tns:pnode name="cdnode*" pattern="wildcard">
          <tns:user name="MUSR_.*"
            ignorecase="true"
            pattern="regex"
            cdUserId="bob"
            cdPassword="passw0rd"
            pnodeUserId="bill"
            pnodePassword="alacazam">
          <tns:snode name="cdnode2" pattern="wildcard" userId="sue" password="foo"/>
          </tns:user>
        </tns:pnode>
      </tns:agent>
    </tns:credentials>

    -->

    <element name="credentials" type="tns:credentialsType"/>

    <complexType name="credentialsType">
      <sequence>
        <element name="agent" type="tns:agentType" minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
    </complexType>

    <complexType name="agentType">
      <sequence>
        <element name="pnode" type="tns:pnodeType" minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="name" type="string" use="required"/>
    </complexType>

    <complexType name="pnodeType">
      <sequence>
        <element name="user" type="tns:userType" minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="name" type="string" use="required"/>
      <attribute name="pattern" type="tns:patternType" use="optional"/>
  -->

```

```

</complexType>

<complexType name="userType">
  <sequence>
    <element name="snode" type="tns:snodeType" minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
  <attribute name="name" type="string" use="required"/>
  <attribute name="ignorecase" type="boolean" use="optional"/>
  <attribute name="pattern" type="tns:patternType" use="optional"/>
  <attribute name="cdUserId" type="string" use="optional"/>
  <attribute name="cdUserIdCipher" type="string" use="optional"/>
  <attribute name="cdPassword" type="string" use="optional"/>
  <attribute name="cdPasswordCipher" type="string" use="optional"/>
  <attribute name="pnodeUserId" type="string" use="optional"/>
  <attribute name="pnodeUserIdCipher" type="string" use="optional"/>
  <attribute name="pnodePassword" type="string" use="optional"/>
  <attribute name="pnodePasswordCipher" type="string" use="optional"/>
</complexType>

<complexType name="snodeType">
  <attribute name="name" type="string" use="required"/>
  <attribute name="pattern" type="tns:patternType" use="optional"/>
  <attribute name="userId" type="string" use="optional"/>
  <attribute name="userIdCipher" type="string" use="optional"/>
  <attribute name="password" type="string" use="optional"/>
  <attribute name="passwordCipher" type="string" use="optional"/>
</complexType>

<simpleType name="patternType">
  <restriction base="string">
    <enumeration value="regex"/>
    <enumeration value="wildcard"/>
  </restriction>
</simpleType>
</schema>

```

Description du fichier ConnectDirectCredentials.xml

Les éléments et attributs utilisés dans le fichier ConnectDirectCredentials.xml sont décrits dans la liste suivante.

< données_d'identification >

Élément de groupe contenant des éléments qui décrivent les données d'identification utilisées par un agent de pont Connect:Direct pour se connecter à un noeud Connect:Direct .

< agent >

Élément de groupe contenant des éléments pour les définitions < pnode > d'un agent nommé.

< noeud >

Noeud principal (PNODE) dans le transfert Connect:Direct . Ce noeud lance la connexion au noeud secondaire (SNODE).

Attribut	Description
nom	Nom du noeud Connect:Direct . La valeur de cet attribut peut être un modèle qui correspond à de nombreux noms de noeud.
schéma	Indique le type de modèle utilisé pour la valeur de l'attribut de nom. Les valeurs admises pour l'attribut pattern sont les suivantes: <ul style="list-style-type: none"> • joker-les caractères génériques sont utilisés • regex-des expressions régulières Java sont utilisées

< utilisateur >

Utilisateur IBM MQ qui soumet la demande de transfert.

Attribut	Description
nom	Nom d'utilisateur utilisé avec Managed File Transfer. La valeur de cet attribut peut être un modèle qui correspond à de nombreux noms d'utilisateur.

Attribut	Description
ignorecase	Indique si la casse du nom est ignorée. Les valeurs admises pour l'attribut ignorecase sont les suivantes: <ul style="list-style-type: none"> • true-le nom n'est pas sensible à la casse • false-le nom est sensible à la casse
schéma	Indique le type de modèle utilisé pour la valeur de l'attribut de nom. Les valeurs admises pour l'attribut pattern sont les suivantes: <ul style="list-style-type: none"> • joker-les caractères génériques sont utilisés • regex-des expressions régulières Java sont utilisées
ID cdUserou cdUserIdCipher	Nom d'utilisateur utilisé par le pont Connect:Direct pour se connecter à son noeud Connect:Direct associé. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
Chiffrement cdPassword ou cdPassword	Mot de passe associé au nom d'utilisateur spécifié par l'attribut d'ID cdUser. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
pnodeUserID ou pnodeUserIdCipher	Nom d'utilisateur utilisé par le noeud principal Connect:Direct . Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
Chiffrement pnodePassword ou pnodePassword	Mot de passe associé au nom d'utilisateur spécifié par l'attribut d'ID pnodeUser. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.

< noeuds >

Le noeud Connect:Direct qui joue le rôle de noeud secondaire (SNODE) lors du transfert de fichiers Connect:Direct .

Attribut	Description
nom	Nom du noeud Connect:Direct . La valeur de cet attribut peut être un modèle qui correspond à de nombreux noms de noeud.
schéma	Indique le type de modèle utilisé pour la valeur de l'attribut de nom. Les valeurs admises pour l'attribut pattern sont les suivantes: <ul style="list-style-type: none"> • joker-les caractères génériques sont utilisés • regex-des expressions régulières Java sont utilisées
Chiffrement userId ou userId	Nom d'utilisateur utilisé pour se connecter à ce noeud lors d'un transfert de fichier. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
mot de passe ou passwordCipher	Mot de passe associé au nom d'utilisateur spécifié par l'attribut userId . Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.

Exemple

Dans cet exemple, l'agent de pont Connect:Direct se connecte au Connect:Direct noeud pnode1. Lorsqu'un utilisateur IBM MQ dont le nom d'utilisateur commence par le préfixe fteuser suivi d'un seul caractère, par exemple fteuser12, demande un transfert impliquant le pont Connect:Direct , l'agent de pont Connect:Direct utilise le nom d'utilisateur cduser et le mot de passe passw0rd pour se connecter au Connect:Direct noeud pnode1. Lorsque le Connect:Direct noeud pnode1 effectue sa partie du transfert, il utilise le nom d'utilisateur pnodeuser et le mot de passe passw0rd1.

Si le nom du noeud secondaire dans le transfert Connect:Direct commence par le préfixe FISH, le noeud pnode1 utilise le nom d'utilisateur fishuser et le mot de passe passwd2 pour se connecter au noeud secondaire. Si le nom du noeud secondaire dans le transfert Connect:Direct commence par le préfixe CHIPS, le noeud pnode1 utilise le nom d'utilisateur chipsuser et le mot de passe passwd3 pour se connecter au noeud secondaire.

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:credentials xmlns:tns="http://wmqfte.ibm.com/ConnectDirectCredentials"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ConnectDirectCredentials
ConnectDirectCredentials.xsd">
  <tns:agent name="CDAGENT01">
    <tns:pnode name="pnode1" pattern="wildcard">
      <tns:user name="fteuser?" pattern="wildcard" ignorecase="true"
        cdUserId="cduser" cdPassword="passwd"
        pnodeUserId="pnodeuser" pnodePassword="passwd1">
      <tns:snode name="FISH*" pattern="wildcard"
        userId="fishuser" password="passwd2"/>
      <tns:snode name="CHIPS*" pattern="wildcard"
        userId="chipsuser" password="passwd3"/>
    </tns:user>
  </tns:pnode>
</tns:agent>
</tns:credentials>
```

Format du fichier de propriétés du noeud Connect:Direct

Le fichier ConnectDirectNodeProperties.xml du répertoire de configuration de l'agent de pont Connect:Direct spécifie des informations sur les noeuds Connect:Direct distants impliqués dans un transfert de fichiers.

Le fichier ConnectDirectNodeProperties.xml doit être conforme au schéma ConnectDirectNodeProperties.xsd. Le document de schéma ConnectDirectNodeProperties.xsd se trouve dans le répertoire *MQ_INSTALLATION_PATH/mqft/samples/schema* de l'installation MFT. Un fichier modèle ConnectDirectNodeProperties.xml est créé par la commande **fteCreateCDAgent** dans le répertoire de configuration de l'agent.

Le fichier ConnectDirectNodeProperties.xml est rechargé périodiquement par l'agent et toute modification valide apportée au fichier aura une incidence sur le comportement de l'agent. L'intervalle de rechargement par défaut est de 30 secondes. Cet intervalle peut être modifié en spécifiant la propriété d'agent xmlConfigReloadInterval dans le fichier agent.properties.

Schéma

Le schéma suivant décrit les éléments valides dans le fichier ConnectDirectNodeProperties.xml.

```
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="http://wmqfte.ibm.com/ConnectDirectNodeProperties"
  elementFormDefault="qualified"
  xmlns="https://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://wmqfte.ibm.com/ConnectDirectNodeProperties">
  <element name="nodeProperties" type="tns:nodePropertiesType"></element>
  <complexType name="nodePropertiesType">
    <sequence>
      <element name="credentialsFile" type="tns:credentialsFileName" minOccurs="0" maxOccurs="1"/>
      <element name="node" type="tns:nodeType" minOccurs="0" maxOccurs="unbounded"></element>
    </sequence>
  </complexType>
  <complexType name="nodeType">
    <attribute name="name" type="string" use="required"/>
    <attribute name="pattern" type="tns:patternType" use="optional"/>
    <attribute name="type" type="string" use="required"/>
  </complexType>
  <simpleType name="patternType">
    <restriction base="string">
      <enumeration value="regex"/>
    </restriction>
  </simpleType>
```

```

    <enumeration value="wildcard"/>
  </restriction>
</simpleType>
</schema>

```

Description du fichier ConnectDirectNodeProperties.xml

Les éléments et attributs utilisés dans le fichier ConnectDirectNodeProperties.xml sont décrits dans la liste suivante.

nodeProperties

Élément racine du document XML.

credentialsFile

Chemin d'accès au fichier de données d'identification dans lequel les informations sensibles sont stockées. Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir [«Utilisation des variables d'environnement dans les propriétés MFT»](#), à la page 547

noeud

Indique un ou plusieurs noeuds Connect:Direct .

Attribut	Description
nom	Modèle qui identifie les noms des noeuds Connect:Direct qui utilisent les définitions spécifiées par l'élément node. La correspondance de modèle n'est pas sensible à la casse.
schéma	Indique le type de modèle utilisé pour la valeur de l'attribut name . Les valeurs admises pour l'attribut pattern sont les suivantes: <ul style="list-style-type: none"> • joker-les caractères génériques sont utilisés • regex-des expressions régulières Java sont utilisées Pour plus d'informations sur les types d'expression régulière utilisés par MFT, voir «Expressions régulières utilisées par MFT» , à la page 679.
type	Indique le type de système d'exploitation du ou des noeuds Connect:Direct qui correspondent au modèle indiqué par l'attribut name . Les valeurs admises pour l'attribut type sont les suivantes: <ul style="list-style-type: none"> • Windows -le noeud s'exécute sur Windows • UNIX -le noeud s'exécute sous UNIX ou Linux •  z/OS, zos, os/390ou os390 -le noeud s'exécute sur z/OS La valeur de cet attribut n'est pas sensible à la casse.

Exemple

Dans cet exemple, le nom de fichier des données d'identification Connect:Direct est spécifié sous la forme ConnectDirectCredentials.xml. L'exemple de code spécifie les connexions de plateforme suivantes:

- Tous les noeuds Connect:Direct dont le nom commence par "cdnodew" s'exécutent sur la plateforme Windows .
- Tous les noeuds Connect:Direct dont le nom commence par "cdnodeu" s'exécutent sur la plateforme UNIX .
- Tous les noeuds Connect:Direct dont le nom commence par "cdnodez" s'exécutent sur la plateforme z/OS .
- Tous les autres noeuds Connect:Direct s'exécutent sur la plateforme UNIX .

L'agent de pont Connect:Direct recherche les correspondances entre le début et la fin du fichier et utilise la première correspondance qu'il trouve.

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:nodeProperties xmlns:tns="http://wmqfte.ibm.com/ConnectDirectNodeProperties"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ConnectDirectNodeProperties
    ConnectDirectNodeProperties.xsd">

  <tns:credentialsFile path="ConnectDirectCredentials.xml"/>
  <tns:node name="cdnodew*" pattern="wildcard" type="windows"/>
  <tns:node name="cdnodeu.*" pattern="regex" type="unix"/>
  <tns:node name="cdnodez*" pattern="wildcard" type="zos"
  <tns:node name="*" pattern="wildcard" type="unix"/>

</tns:nodeProperties>
```

Format de fichier des définitions de processus Connect:Direct

Le fichier ConnectDirectProcessDefinitions.xml du répertoire de configuration de l'agent de pont Connect:Direct spécifie le processus Connect:Direct défini par l'utilisateur à démarrer dans le cadre du transfert de fichiers.

Le fichier ConnectDirectProcessDefinitions.xml doit être conforme au schéma ConnectDirectProcessDefinitions.xsd. Le document de schéma ConnectDirectProcessDefinitions.xsd se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/schema` de l'installation MFT. Un fichier modèle ConnectDirectProcessDefinitions.xml est créé par la commande **fteCreateCDAgent** dans le répertoire de configuration de l'agent.

Le fichier ConnectDirectProcessDefinitions.xml est rechargé périodiquement par l'agent et toute modification valide apportée au fichier aura une incidence sur le comportement de l'agent. L'intervalle de rechargement par défaut est de 30 secondes. Cet intervalle peut être modifié en spécifiant la propriété d'agent `xmlConfigReloadInterval` dans le fichier `agent.properties`.

Schéma

Le schéma suivant décrit les éléments valides dans le fichier ConnectDirectProcessDefinitions.xml.

```
<schema targetNamespace="http://wmqfte.ibm.com/ConnectDirectProcessDefinitions"
  elementFormDefault="qualified"
  xmlns="https://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://wmqfte.ibm.com/ConnectDirectProcessDefinitions">

  <element name="cdprocess" type="tns:cdprocessType"></element>

  <complexType name="cdprocessType">
    <sequence>
      <element name="processSet" type="tns:processSetType"
        minOccurs="0" maxOccurs="unbounded"></element>
    </sequence>
  </complexType>

  <complexType name="processSetType">
    <sequence>
      <element name="condition" type="tns:conditionType"
        minOccurs="0" maxOccurs="1"/>
      <element name="process" type="tns:processType"
        minOccurs="1" maxOccurs="1"/>
    </sequence>
  </complexType>

  <complexType name="conditionType">
    <choice minOccurs="0" maxOccurs="unbounded">
      <element name="match" type="tns:matchType"/>
      <element name="defined" type="tns:definedType"/>
    </choice>
  </complexType>

  <complexType name="matchType">
    <attribute name="variable" type="string" use="required"/>
  </complexType>
```

```

    <attribute name="value" type="string" use="required"/>
    <attribute name="pattern" type="tns:patternType" use="optional"/>
  </complexType>

  <complexType name="definedType">
    <attribute name="variable" type="string" use="required"/>
  </complexType>

  <complexType name="processType">
    <sequence>
      <element name="preTransfer" type="tns:transferType"
        minOccurs="0" maxOccurs="1"/>
      <element name="transfer" type="tns:transferType"
        minOccurs="0" maxOccurs="1"/>
      <element name="postTransferSuccess" type="tns:transferType"
        minOccurs="0" maxOccurs="1"/>
      <element name="postTransferFailure" type="tns:transferType"
        minOccurs="0" maxOccurs="1"/>
    </sequence>
  </complexType>

  <complexType name="transferType">
    <attribute name="process" type="string" use="required"/>
  </complexType>

  <simpleType name="patternType">
    <restriction base="string">
      <enumeration value="regex"/>
      <enumeration value="wildcard"/>
    </restriction>
  </simpleType>

</schema>

```

Description du fichier ConnectDirectProcessDefinitions.xml

Les éléments et attributs utilisés dans le fichier ConnectDirectProcessDefinitions.xml sont décrits dans la liste suivante.

cdProcess

Élément racine du document XML.

processSet

Élément de groupe contenant toutes les informations sur un ensemble de processus définis par l'utilisateur.

condition

Élément de groupe contenant les conditions par rapport auxquelles un transfert est testé pour déterminer si l'ensemble de processus contenu dans l'élément processSet est utilisé.

correspond à

Condition qui teste si la valeur d'une variable correspond à une valeur donnée.

Attribut	Description
variable	Indique une variable. La valeur de cette variable est comparée à la valeur de l'attribut value . La variable est un symbole intrinsèque. Pour plus d'informations, voir la section «Variables de substitution à utiliser avec les processus Connect:Direct définis par l'utilisateur», à la page 680.
valeur	Indique un modèle à mettre en correspondance avec la valeur de la variable spécifiée par l'attribut variable .
schéma	Indique le type de modèle utilisé pour la valeur de l'attribut value . Les valeurs admises pour l'attribut pattern sont les suivantes: <ul style="list-style-type: none"> • wildcard -les caractères génériques sont utilisés • regex -Les expressions régulières Java sont utilisées Cet attribut est facultatif et la valeur par défaut est wildcard.

défini

Condition qui teste si une variable a été définie.

Attribut	Description
variable	Indique une variable. Si cette variable existe, la condition de correspondance est satisfaite. La variable est un symbole intrinsèque. Pour plus d'informations, voir la section «Variables de substitution à utiliser avec les processus Connect:Direct définis par l'utilisateur», à la page 680.

processus

Élément de groupe contenant les informations sur l'emplacement des processus Connect:Direct à appeler lorsqu'une correspondance est trouvée.

transfert

Processus Connect:Direct à appeler lors d'une demande de transfert.

Attribut	Description
processus	Facultatif. Indique le nom d'un fichier contenant un processus Connect:Direct à appeler lors d'une demande de transfert. Le chemin d'accès au fichier est relatif au répertoire de configuration de l'agent de pont Connect:Direct . Cet attribut est facultatif. Par défaut, un processus généré par MFTTest utilisé. Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.

Exemple

Dans cet exemple, il existe trois éléments processSet .

Le premier élément processSet indique que si une demande de transfert comporte une variable **%FTESNODE** dont la valeur correspond au modèle `Client*` et une variable **%FTESUSER** dont la valeur est `Admin`, l'agent de pont Connect:Direct soumet le processus Connect:Direct situé dans le `agent_configuration_directory/AdminClient.cdp` dans le cadre du transfert.

Le second élément processSet indique que si une demande de transfert comporte une variable **%FTESNODE** avec une valeur qui correspond au modèle `Client*`, l'agent de pont Connect:Direct soumet le processus Connect:Direct situé dans `agent_configuration_directory/Client.cdp` dans le cadre du transfert. L'agent de pont Connect:Direct lit les éléments processSet dans l'ordre dans lequel ils sont définis, et s'il trouve une correspondance, il utilise la première correspondance et ne recherche pas une autre correspondance. Pour les demandes de transfert qui correspondent aux conditions du premier et du second processSet, l'agent de pont Connect:Direct appelle uniquement les processus spécifiés par le premier processSet.

Le troisième élément processSet n'a pas de conditions et correspond à tous les transferts. Si la demande de transfert ne correspond pas aux conditions de la première ou de la deuxième processSet, l'agent de pont Connect:Direct soumet le processus Connect:Direct spécifié par la troisième condition. Ce processus se trouve dans le fichier `agent_configuration_directory/Default.cdp` dans le cadre du transfert.

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:cdprocess xmlns:tns="http://wmqfte.ibm.com/ConnectDirectProcessDefinitions"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ConnectDirectProcessDefinitions
ConnectDirectProcessDefinitions.xsd">
  <tns:processSet>
    <tns:condition>
      <tns:match variable="%FTESNODE" value="Client*" pattern="wildcard"/>
      <tns:match variable="%FTESUSER" value="Admin" pattern="wildcard"/>
    </tns:condition>
    <tns:process>
      <tns:transfer process="AdminClient.cdp"/>
    </tns:process>
  </tns:processSet>
  <tns:processSet>
    <tns:process>
      <tns:transfer process="Client.cdp"/>
    </tns:process>
  </tns:processSet>
  <tns:processSet>
    <tns:process>
      <tns:transfer process="Default.cdp"/>
    </tns:process>
  </tns:processSet>
</tns:cdprocess>
```

```

    </tns:process>
</tns:processSet>

<tns:processSet>
  <tns:condition>
    <tns:match variable="%FTESNODE" value="Client*" pattern="wildcard" />
  </tns:condition>
  <tns:process>
    <tns:transfer process="Client.cdp" />
  </tns:process>
</tns:processSet>

<tns:processSet>
  <tns:process>
    <tns:transfer process="Default.cdp" />
  </tns:process>
</tns:processSet>
</tns:cdprocess>

```

Utilisation des bacs à sable utilisateur MFT

Vous pouvez restreindre la zone du système de fichiers dans laquelle les fichiers peuvent être transférés en fonction du nom d'utilisateur MQMD qui demande le transfert.

Les bacs à sable utilisateur ne sont pas pris en charge lorsque l'agent est un agent de pont de protocole ou un agent de pont Connect:Direct .

Pour activer le bac à sable utilisateur, ajoutez la propriété suivante au fichier `agent.properties` de l'agent que vous souhaitez restreindre:

```
userSandboxes=true
```

Lorsque cette propriété est présente et définie sur `true`, l'agent utilise les informations du fichier `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name/UserSandboxes.xml` pour déterminer les parties du système de fichiers auxquelles l'utilisateur qui demande le transfert peut accéder.

Le XML `UserSandboxes.xml` est composé d'un élément `<agent>` qui contient zéro ou plusieurs éléments `<sandbox>` . Ces éléments décrivent quelles règles sont appliquées à quels utilisateurs. L'attribut `user` de l'élément `<sandbox>` est un modèle utilisé pour établir une correspondance avec l'utilisateur MQMD de la demande.

Le fichier `UserSandboxes.xml` est rechargé périodiquement par l'agent et toute modification valide apportée au fichier aura une incidence sur le comportement de l'agent. L'intervalle de rechargement par défaut est de 30 secondes. Vous pouvez modifier cet intervalle en spécifiant la propriété d'agent `xmlConfigReloadInterval` dans le fichier `agent.properties` .

Si vous spécifiez l'attribut ou la valeur `userPattern="regex"` , l'attribut `user` est interprété comme une expression régulière Java . Pour plus d'informations, voir la section «[Expressions régulières utilisées par MFT](#)», à la page 679.

Si vous ne spécifiez pas l'attribut ou la valeur `userPattern="regex"` , l'attribut `user` est interprété comme un modèle avec les caractères génériques suivants:

- astérisque (*), qui représente zéro ou plusieurs caractères
- point d'interrogation (?), qui représente exactement un caractère

Les correspondances sont effectuées dans l'ordre dans lequel les éléments `<sandbox>` sont répertoriés dans le fichier. Seule la première correspondance est utilisée, toutes les correspondances potentielles suivantes dans le fichier sont ignorées. Si aucun des éléments `<sandbox>` spécifiés dans le fichier ne correspond à l'utilisateur MQMD associé au message de demande de transfert, le transfert ne peut pas accéder au système de fichiers. Lorsqu'une correspondance a été trouvée entre le nom d'utilisateur MQMD et un attribut `user` , la correspondance identifie un ensemble de règles dans un élément `<sandbox>` qui sont appliquées au transfert. Cet ensemble de règles est utilisé pour déterminer quels fichiers, ou ensembles de données, peuvent être lus ou écrits dans le cadre du transfert.

Chaque ensemble de règles peut spécifier un élément `<read>` , qui identifie les fichiers qui peuvent être lus, et un élément `<write>` , qui identifie les fichiers qui peuvent être écrits. Si vous omettez les éléments `<read>` ou `<write>` d'un ensemble de règles, il est supposé que l'utilisateur associé à cet ensemble de règles n'est pas autorisé à effectuer des lectures ou des écritures, selon le cas.

Remarque : L'élément `<read>` doit être antérieur à l'élément `<write>` et l'élément `<include>` doit être antérieur à l'élément `<exclude>` , dans le fichier `UserSandboxes.xml` .

Chaque élément `<read>` ou `<write>` contient un ou plusieurs canevas utilisés pour déterminer si un fichier se trouve dans le bac à sable et peut être transféré. Spécifiez ces modèles à l'aide des éléments `<include>` et `<exclude>` . L'attribut `name` de l'élément `<include>` ou `<exclude>` spécifie le modèle à mettre en correspondance. Un attribut `type` facultatif indique si la valeur de nom est un fichier ou un modèle de file d'attente. Si l'attribut `type` n'est pas spécifié, l'agent traite le modèle comme un modèle de chemin de fichier ou de répertoire. Exemple :

```
<tns:read>
  <tns:include name="/home/user/**"/>
  <tns:include name="USER.**" type="queue"/>
  <tns:exclude name="/home/user/private/**"/>
</tns:read>
```

Les modèles `<include>` et `<exclude>` `name` sont utilisés par l'agent pour déterminer si les fichiers, les ensembles de données, ou les files d'attente peuvent être lus ou écrits. Une opération est autorisée si le chemin de fichier canonique, l'ensemble de données ou le nom de file d'attente correspond à au moins un des modèles inclus et à exactement zéro des modèles exclus. Les modèles spécifiés à l'aide de l'attribut `name` des éléments `<include>` et `<exclude>` utilisent les séparateurs de chemin et les conventions appropriés à la plateforme sur laquelle l'agent s'exécute. Si vous spécifiez des chemins de fichier relatifs, les chemins sont résolus par rapport à la propriété `transferRoot` de l'agent.

Lors de la spécification d'une restriction de file d'attente, la syntaxe `QUEUE@QUEUEMANAGER` est prise en charge, avec les règles suivantes:

- Si le caractère at (@) est manquant dans l'entrée, le modèle est traité comme un nom de file d'attente accessible sur n'importe quel gestionnaire de files d'attente. Par exemple, si le modèle est `name` , il est traité de la même manière que `name@**`.
- Si le caractère arobase (@) est le premier caractère de l'entrée, le modèle est traité comme un nom de gestionnaire de files d'attente et toutes les files d'attente du gestionnaire de files d'attente sont accessibles. Par exemple, si le modèle est `@name` , il est traité de la même manière que `**@name`.

Les caractères génériques suivants ont une signification spéciale lorsque vous les spécifiez dans le cadre de l'attribut `name` des éléments `<include>` et `<exclude>` :

Un astérisque unique correspond à zéro ou plusieurs caractères dans un nom de répertoire ou dans un qualificateur d'un nom de fichier ou d'un nom de file d'attente .

?

Un point d'interrogation correspond exactement à un caractère dans un nom de répertoire ou dans un qualificateur d'un nom de fichier ou d'un nom de file d'attente .

Deux astérisques correspondent à zéro ou plusieurs noms de répertoire, ou à zéro ou plusieurs qualificatifs dans un nom de fichier ou un nom de file d'attente . En outre, les chemins qui se terminent par un séparateur de chemin ont un `"**"` implicite ajouté à la fin du chemin. Ainsi, `/home/user/` est identique à `/home/user/**`.

Exemple :

- `/**/test/**` correspond à tout fichier dont le chemin contient un répertoire `test`
- `/test/file?` correspond à tout fichier du répertoire `/test` qui commence par la chaîne `file` suivie d'un caractère unique

- `c:\test*.txt` correspond à tout fichier du répertoire `c:\test` avec une extension `.txt`
- `c:\test***.txt` correspond à n'importe quel fichier du répertoire `c:\test` ou à l'un de ses sous-répertoires dont l'extension est `.txt`
-  `// 'TEST.*.DATA'` correspond à tout fichier dont le premier qualificateur est TEST, dont le second est un qualificateur et dont le troisième est DATA.
- `*@QM1` correspond à toute file d'attente du gestionnaire de files d'attente QM1 comportant un qualificateur unique.
- `TEST.*.QUEUE@QM1` correspond à n'importe quelle file d'attente du gestionnaire de files d'attente QM1 qui possède le premier qualificateur TEST, un deuxième qualificateur et un troisième qualificateur QUEUE.
- `**@QM1` correspond à n'importe quelle file d'attente du gestionnaire de files d'attente QM1.

Liens symboliques

Vous devez résoudre complètement tous les liens symboliques que vous utilisez dans les chemins de fichier du fichier `UserSandboxes.xml` en spécifiant des liens fixes dans les éléments `<include>` et `<exclude>`. Par exemple, si vous disposez d'un lien symbolique dans lequel `/var` est mappé à `/SYSTEM/var`, vous devez spécifier ce chemin en tant que `<tns:include name="/SYSTEM/var"/>`, sinon le transfert prévu échoue avec une erreur de sécurité du bac à sable de l'utilisateur.

Exemple

Cet exemple montre comment autoriser l'utilisateur avec le nom d'utilisateur MQMD guest à transférer un fichier depuis le répertoire `/home/user/public` ou l'un de ses sous-répertoires sur le système où l'agent AGENT_JUPITER est en cours d'exécution, en ajoutant l'élément `<sandbox>` suivant au fichier `UserSandboxes.xml` dans le répertoire de configuration d'AGENT_JUPITER:

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:userSandboxes
  xmlns:tns="http://wmqfte.ibm.com/UserSandboxes"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/UserSandboxes UserSandboxes.xsd">
  <tns:agent>
    <tns:sandbox user="guest">
      <tns:read>
        <tns:include name="/home/user/public/**"/>
      </tns:read>
    </tns:sandbox>
  </tns:agent>
</tns:userSandboxes>
```

Exemple

Cet exemple montre comment autoriser tout utilisateur avec le nom d'utilisateur MQMD account suivi d'un chiffre unique, par exemple `account4`, à effectuer les actions suivantes:

- Transférez tout fichier à partir du répertoire `/home/account` ou de l'un de ses sous-répertoires, à l'exception du répertoire `/home/account/private` sur le système où l'agent AGENT_SATURN est en cours d'exécution
- Transférez tout fichier dans le répertoire `/home/account/output` ou dans l'un de ses sous-répertoires sur le système où l'agent AGENT_SATURN est en cours d'exécution.
- Lire les messages des files d'attente du gestionnaire de files d'attente local en commençant par le préfixe `ACCOUNT.`, sauf s'il commence par `ACCOUNT.PRIVATE.` (c'est-à-dire avec `PRIVATE` au deuxième niveau).
- Transférez les données dans les files d'attente en commençant par le préfixe `ACCOUNT.OUTPUT.` sur n'importe quel gestionnaire de files d'attente.

Pour permettre à un utilisateur doté du nom d'utilisateur MQMD account d'effectuer ces actions, ajoutez l'élément `<sandbox>` suivant au fichier `UserSandboxes.xml`, dans le répertoire de configuration d'AGENT_SATURN:

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:userSandboxes
  xmlns:tns="http://wmqfte.ibm.com/UserSandboxes"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/UserSandboxes UserSandboxes.xsd">
  <tns:agent>
    <tns:sandbox user="account[0-9]" userPattern="regex">
      <tns:read>
        <tns:include name="/home/account/**"/>
        <tns:include name="ACCOUNT.**" type="queue"/>
        <tns:exclude name="ACCOUNT.PRIVATE.**" type="queue"/>
        <tns:exclude name="/home/account/private/**"/>
      </tns:read>
      <tns:write>
        <tns:include name="/home/account/output/**"/>
        <tns:include name="ACCOUNT.OUTPUT.**" type="queue"/>
      </tns:write>
    </tns:sandbox>
  </tns:agent>
</tns:userSandboxes>
```

Référence associée

«Vérifications supplémentaires pour les transferts de caractères génériques», à la page 51

Depuis IBM MQ 9.0.1 dans Continuous Delivery et IBM MQ 9.0.0 Fix Pack 1 dans Long Term Support, si un agent a été configuré avec un bac à sable d'utilisateur ou d'agent afin de restreindre les emplacements vers et depuis lesquels l'agent peut transférer des fichiers, vous pouvez spécifier que des vérifications supplémentaires doivent être effectuées sur les transferts de caractères génériques pour cet agent.

«Utilisation des bacs à sable utilisateur MFT», à la page 48

Vous pouvez restreindre la zone du système de fichiers dans laquelle les fichiers peuvent être transférés en fonction du nom d'utilisateur MQMD qui demande le transfert.

«Le fichier `MFT.agent.properties`», à la page 564

Chaque Managed File Transfer Agent possède son propre fichier de propriétés, `agent.properties`, qui doit contenir les informations qu'un agent utilise pour se connecter à son gestionnaire de files d'attente. Le fichier `agent.properties` peut également contenir des propriétés qui modifient le comportement de l'agent.

Propriétés de configuration du consignateur MFT

Le consignateur Managed File Transfer possède un ensemble de propriétés de configuration. Spécifiez ces propriétés dans le fichier `logger.properties`, qui se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

Pour IBM WebSphere MQ 7.5 ou version ultérieure, il est possible d'utiliser des variables d'environnement dans certaines propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction des changements d'environnement, tels que l'utilisateur qui exécute le processus. Pour plus d'informations, voir la section «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.

Remarque : Lorsque vous spécifiez des chemins de fichier sous Windows, le caractère de séparation barre oblique inversée (`\`) doit apparaître sous la forme de doubles barres obliques inversées (`\\`) (c'est-à-dire, avec caractères d'échappement `\`). Vous pouvez également utiliser une barre oblique (`/`) comme séparateur. Pour plus d'informations sur l'échappement des caractères dans les fichiers de propriétés Java dans Oracle, voir [Javadoc pour la classe Properties](#).

Propriétés de connexion en mode liaisons

Tableau 47. Propriétés de connexion pour le mode liaisons		
Nom de la propriété	Description	Valeur par défaut
wmqfte.logger.type	Type de consignateur utilisé: fichier ou base de données. Définissez cette valeur sur FILE ou DATABASE.	Aucune valeur par défaut
wmqfte.max.transaction.messages	Nombre maximal de messages traités dans une transaction avant la validation de la transaction. En mode de journalisation avec réutilisation automatique des journaux, un gestionnaire de files d'attente dispose d'une quantité fixe d'espace disponible pour les données en cours. Veuillez à définir cette propriété avec une valeur suffisamment faible pour que l'espace disponible ne soit pas saturé.	50
wmqfte.max.transaction.time	Durée maximale, en millisecondes, qui s'écoule entre les validations de transaction.	5000
wmqfte.max.consecutive.reject	Nombre maximal de messages pouvant être rejetés consécutivement (c'est-à-dire sans rencontrer de message valide). Si ce nombre est dépassé, le consignateur conclut que le problème n'est pas lié aux messages eux-mêmes, mais à la configuration. Par exemple, si vous créez une colonne de nom d'agent dans la base de données plus étroite que tous les noms d'agent, tous les messages faisant référence aux agents sont rejetés.	50
wmqfte.reject.queue.name	Nom d'une file d'attente dans laquelle le consignateur insère des messages qu'il ne peut pas traiter. Si vous disposez d'un consignateur de base de données, voir Traitement et rejet des erreurs du consignateur de base de données pour plus de détails sur les messages pouvant être placés dans cette file d'attente.	SYSTEME SYSTEM.FTE.LOG.RJCT. <i>nom_connexion</i>
wmqfte.command.queue.name	Nom d'une file d'attente à partir de laquelle le consignateur lit les messages de commande qui contrôlent son comportement.	SYSTEME SYSTEM.FTE.LOG.CMD. <i>nom_journal</i>
wmqfte.queue.manager	Gestionnaire de files d'attente auquel le consignateur se connecte. Ce paramètre est obligatoire et correspond à tout ce qui est nécessaire pour les connexions en mode liaison au gestionnaire de files d'attente. (Pour les propriétés de connexion à un gestionnaire de files d'attente éloignées, voir Tableau 48, à la page 620.)	Aucune valeur par défaut

Tableau 47. Propriétés de connexion pour le mode liaisons (suite)

Nom de la propriété	Description	Valeur par défaut
wmqfte.message.source.type	<p>L'une de valeurs suivantes :</p> <p>Abonnement automatique Valeur par défaut. Le consignateur crée et utilise son propre abonnement durable et géré sur le gestionnaire de files d'attente défini dans SYSTEM.FTE/Log/#. Il s'agit d'une valeur appropriée pour la plupart des scénarios.</p> <p>abonnement d'administration Si l'abonnement automatique n'est pas approprié, vous pouvez définir un autre abonnement (par exemple, à l'aide de IBM MQ Explorer, MQSC ou PCF) et demander au consignateur d'utiliser cet abonnement. Par exemple, utilisez cette valeur pour partitionner l'espace journal de sorte qu'un consignateur gère les agents de A-H, un autre consignateur gère I-P et un troisième consignateur de Q-Z.</p> <p>file d'attente Si la topologie IBM MQ signifie que la création d'un abonnement pour le consignateur n'est pas pratique, vous pouvez utiliser une file d'attente à la place. Configurez IBM MQ de sorte que la file d'attente reçoive les messages qui sont généralement reçus par un abonnement à SYSTEM.FTE/Log/# sur le gestionnaire de file d'attente de coordination.</p>	Abonnement automatique
wmqfte.message.source.name	Si le type de source de messages est abonnement d'administration ou file d'attente, nom de l'abonnement ou de la file d'attente à utiliser. Cette propriété est ignorée si le type de source est abonnement automatique.	Aucune valeur par défaut
wmqfte.database.credentials.file	<p>Fichier contenant le nom d'utilisateur et le mot de passe pour la connexion à la base de données.</p> <p>Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.</p> <p>Pour plus d'informations, voir «Format de fichier de données d'identification MFT», à la page 835.</p>	<p> Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir Création d'un fichier de données d'identification MFT</p> <p> Pour plus d'informations sur l'emplacement et les droits d'accès de ce fichier, voir Configuration de MQMFTCredentials.xml .</p> <p> Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir MFT et IBM MQ authentification de la connexion .</p>

Tableau 47. Propriétés de connexion pour le mode liaisons (suite)

Nom de la propriété	Description	Valeur par défaut
<p>wmqfte.database.driver</p>	<p>Emplacement des classes du pilote JDBC pour la base de données. Il s'agit généralement du chemin et du nom d'un fichier JAR. Par exemple, le pilote de type 2 pour Db2 sur les systèmes AIX requiert le fichier /opt/IBM/db2/V9.5/java/db2jcc.jar. Sur les systèmes Windows, spécifiez le séparateur de chemin sous la forme d'une barre oblique (/), par exemple, C:/Program Files/IBM/SQLLIB/java/db2jcc.jar. z/OS Sous z/OS, indiquez le chemin d'accès complet du fichier db2jcc.jar. Par exemple, <code>wmqfte.database.driver=/db2/db2v10/jdbc/classes/db2jcc.jar</code>.</p> <p>z/OS Sur les systèmes z/OS, vous devez référencer tous les fichiers JAR suivants:</p> <ul style="list-style-type: none"> • db2jcc.jar • db2jcc_license_cisuz.jar • db2jcc_javax.jar <p>Si votre pilote de base de données se compose de plusieurs fichiers JAR (par exemple, Db2 9.1 requiert un fichier JAR de pilote et un fichier JAR de licence), incluez tous ces fichiers JAR dans cette propriété. Séparez plusieurs noms de fichier à l'aide du séparateur de chemin d'accès aux classes de votre plateforme, c'est-à-dire le point-virgule (;) sur les systèmes Windows et le signe deux-points (:) sur les autres plateformes.</p>	<p>Aucune valeur par défaut</p>
<p>wmqfte.database.exclude_metadata</p>	<p>Contrôle si les entrées sont stockées dans la table de métadonnées qui contient des informations pouvant être trouvées dans d'autres tables du schéma du consigneur de base de données. Définissez cette valeur sur <code>true</code> ou sur <code>false</code>. Ces entrées de métadonnées ne sont plus stockées par défaut car il s'agit d'une duplication des données existantes et d'un gaspillage de la capacité de stockage de la base de données. Les entrées de propriété et les tables dans lesquelles les mêmes données apparaissent sont les suivantes:</p> <ul style="list-style-type: none"> • com.ibm.wmqfte.SourceAgent TRANSFER_EVENT ou CALL_REQUEST • com.ibm.wmqfte.DestinationAgent TRANSFER_EVENT • com.ibm.wmqfte.MqmdUser TRANSFER_EVENT ou CALL_REQUEST • com.ibm.wmqfte.OriginatingUser TRANSFER_EVENT ou CALL_REQUEST • com.ibm.wmqfte.OriginatingHost TRANSFER_EVENT ou CALL_REQUEST • com.ibm.wmqfte.TransferId TRANSFER ou CALL_REQUEST • com.ibm.wmqfte.JobName TRANSFER ou CALL_REQUEST <p>La définition de la valeur de cette propriété sur <code>false</code> entraîne le stockage de ces entrées de métadonnées dans la table de métadonnées.</p>	<p>conforme</p>

Tableau 47. Propriétés de connexion pour le mode liaisons (suite)

Nom de la propriété	Description	Valeur par défaut
wmqfte.database.host	<p>Db2 uniquement :</p> <p>Pour IBM WebSphere MQ 7.5 ou version ultérieure, nom d'hôte du serveur de base de données auquel se connecter à l'aide d'un pilote JDBC de type 4. Si une valeur est spécifiée pour cette propriété, une valeur pour <code>wmqfte.database.port</code> doit également être spécifiée. Si les deux propriétés ne sont pas définies, le consigneur de base de données se connecte à l'aide du pilote JDBC de type 2 par défaut.</p> <p>Si une valeur est spécifiée pour cette propriété, un fichier de données d'identification pour ce consigneur (chemin d'accès au fichier défini par la propriété <code>wmqfte.database.credentials.file</code>) doit exister et être accessible pour définir le nom d'utilisateur et le mot de passe de connexion à la base de données, même si la base de données se trouve sur le système local.</p>	Aucune valeur par défaut
wmqfte.database.name	Nom de l'instance de base de données (ou du sous-système lors de l'utilisation de Db2 for z/OS) qui contient les tables de journal Managed File Transfer .	Aucune valeur par défaut
wmqfte.database.type	Système de gestion de base de données en cours d'utilisation: Db2 ou Oracle. Définissez cette valeur sur <code>db2</code> ou <code>oracle</code> .	db2
wmqfte.database.port	<p>Db2 uniquement :</p> <p>Pour IBM WebSphere MQ 7.5 ou version ultérieure, numéro de port du serveur de base de données auquel se connecter à l'aide d'un pilote JDBC de type 4. Si une valeur est spécifiée pour cette propriété, une valeur pour <code>wmqfte.database.host</code> doit également être spécifiée. Si les deux propriétés ne sont pas définies, le consigneur de base de données se connecte à l'aide du pilote JDBC de type 2 par défaut.</p> <p>Si une valeur est spécifiée pour cette propriété, un fichier de données d'identification pour ce consigneur (chemin d'accès au fichier défini par la propriété <code>wmqfte.database.credentials.file</code>) doit exister et être accessible pour définir le nom d'utilisateur et le mot de passe de connexion à la base de données, même si la base de données se trouve sur le système local.</p>	Aucune valeur par défaut
wmqfte.database.schema	Schéma de base de données qui contient les tables de journalisation Managed File Transfer . Dans la plupart des cas, la valeur par défaut est appropriée, mais vous devrez peut-être spécifier une autre valeur en fonction de vos propres considérations de base de données spécifiques au site.	FTELOG
wmqfte.database.native.library.path	<p>Chemin contenant les bibliothèques natives requises par le pilote de base de données de votre choix (le cas échéant). Par exemple, le pilote de type 2 pour Db2 sur les systèmes AIX requiert des bibliothèques de <code>/opt/IBM/db2/V9.5/lib32/</code>. Comme alternative à cette propriété, vous pouvez définir la propriété système <code>java.library.path</code> à l'aide d'autres méthodes.</p> <p>Sur les systèmes Solaris et HP-UX , avant d'exécuter la commande fteStartLogger , vous devez également définir et exporter la variable d'environnement <code>LD_LIBRARY_PATH</code> pour inclure le chemin.</p>	Aucune valeur par défaut

Tableau 47. Propriétés de connexion pour le mode liaisons (suite)

Nom de la propriété	Description	Valeur par défaut
wmqfte.file.logger.fileDirectory	Répertoire dans lequel se trouvent les fichiers journaux du consignateur de fichiers.	mqft/logs/coordination_dir/loggers/logger_name/logs
wmqfte.file.logger.fileSize	Taille maximale qu'un fichier journal peut atteindre. La valeur de taille est un entier positif, supérieur à zéro, suivi de l'une des unités suivantes: Ko, Mo, Go, m (minutes), h (heures), d (jours), w (semaines). Par exemple, wmqfte.file.logger.fileSize=5MB Spécifie une taille de fichier maximale de 5MB. wmqfte.file.logger.fileSize=2d Indique une taille de fichier maximale de 2 jours de données.	10MB
wmqfte.file.logger.fileCount	Nombre maximal de fichiers journaux à créer. Lorsque la quantité de données dépasse la quantité maximale pouvant être stockée dans ce nombre de fichiers, le fichier le plus ancien est supprimé de sorte que le nombre de fichiers ne dépasse jamais la valeur spécifiée.	3

Tableau 47. Propriétés de connexion pour le mode liaisons (suite)

Nom de la propriété	Description	Valeur par défaut
<p>wmqfte.file.logger.mode</p>	<p>Mode de consignateur utilisé: circulaire ou linéaire. Définissez cette valeur sur CIRCULAIRE ou sur LINEAR.</p> <p>CIRCULAIRE-Le consignateur de fichiers écrit des informations dans un fichier jusqu'à ce que ce dernier atteigne sa taille maximale définie à l'aide de la propriété wmqfte.file.logger.fileSize . Lorsque la taille maximale est atteinte, le consignateur de fichiers démarre un nouveau fichier. Le nombre maximal de fichiers écrits dans ce mode est contrôlé par la valeur définie à l'aide de la propriété wmqfte.file.logger.fileCount . Lorsque ce nombre maximal de fichiers est atteint, le consignateur de fichiers supprime le premier fichier et le recrée pour l'utiliser comme fichier actif. Si la valeur définie dans la propriété wmqfte.file.logger.fileSize est une unité d'octet de taille fixe (par exemple, Ko, Mo ou Go), la limite supérieure de l'espace disque utilisé dans ce mode est égale à fileSize multipliée par fileCount. Si la valeur définie dans la propriété wmqfte.file.logger.fileSize est une unité de temps (par exemple, m, h, dou w), la taille maximale dépend du débit des messages de journal dans votre système au cours de ces périodes. La convention de dénomination des fichiers journaux utilisée lors de l'exécution dans ce mode est: <i>logger_namenumber-timestamp</i>.log où:</p> <ul style="list-style-type: none"> • <i>logger_name</i> est le nom donné au consignateur dans la commande fteCreateLogger . • <i>nombre</i> est le numéro du fichier dans l'ensemble. • <i>horodatage</i> est l'horodatage de la création du fichier. <p>Exemple: LOGGER1-20111216123430147.log</p> <p>LINEAR-Le consignateur de fichiers écrit des informations dans un fichier jusqu'à ce que ce fichier atteigne sa taille maximale définie à l'aide de la propriété wmqfte.file.logger.fileSize . Lorsque la taille maximale est atteinte, le consignateur de fichiers démarre un nouveau fichier. Les fichiers précédemment écrits ne sont pas supprimés, ce qui permet de les conserver en tant qu'enregistrement historique des messages de journal. Les fichiers ne sont pas supprimés lors de l'exécution en mode linéaire . Par conséquent, la propriété wmqfte.file.logger.fileCount est ignorée car il n'y a pas de limite supérieure au nombre de fichiers pouvant être créés. Etant donné qu'il n'y a pas de limite supérieure lors de l'exécution dans ce mode, il est nécessaire de suivre la quantité d'espace disque utilisée par les fichiers journaux afin d'éviter une faible quantité d'espace disque. La convention de dénomination des fichiers journaux utilisée lors de l'exécution dans ce mode est: <i>logger_name-timestamp</i>.log où:</p> <ul style="list-style-type: none"> • <i>logger_name</i> est le nom donné au consignateur dans la commande fteCreateLogger . • <i>horodatage</i> est l'horodatage de la création du fichier. <p>Exemple: LOGGER-20111216123430147.log</p>	<p>Aucune valeur par défaut</p>

Tableau 47. Propriétés de connexion pour le mode liaisons (suite)

Nom de la propriété	Description	Valeur par défaut
wmqfte.max.retry.interval	<p>Durée maximale, en secondes, entre les nouvelles tentatives lorsque le consignateur rencontre une erreur persistante.</p> <p>Certaines conditions d'erreur (par exemple, la perte de la connexion à la base de données) empêchent le consignateur de continuer. Lorsque ce type de condition se produit, le consignateur annule la transaction en cours, attend un certain temps, puis effectue de nouvelles tentatives. Le temps d'attente du consignateur est initialement très court, de sorte que les erreurs transitoires peuvent être surmontées rapidement. Toutefois, à chaque nouvelle tentative du consignateur, le temps d'attente est augmenté. Cela évite trop de travail inutile lorsque la condition d'erreur est plus longue, par exemple lorsqu'une base de données est arrêtée à des fins de maintenance.</p> <p>Utilisez cette propriété pour définir une limite à la durée de l'attente de sorte qu'une nouvelle tentative se produise dans un délai raisonnable après la résolution de la condition d'erreur.</p>	600
loggerQMgrRetryInterval	Intervalle, en secondes, entre les vérifications de la disponibilité du gestionnaire de files d'attente par le contrôleur de processus du consignateur.	30
maxRestartCount	Nombre maximal de redémarrages pouvant se produire dans l'intervalle de temps spécifié par la valeur de la propriété d'intervalle maxRestart. Lorsque cette valeur est dépassée, le contrôleur de processus du consignateur arrête de redémarrer le consignateur et effectue à la place une action basée sur la valeur de la propriété de délai maxRestart.	4
maxRestartInterval	Intervalle, en secondes, pendant lequel le contrôleur de processus du consignateur mesure les redémarrages du consignateur. Si le nombre de redémarrages dans cet intervalle dépasse la valeur de la propriété Nombre maxRestart, le contrôleur de processus du consignateur arrête de redémarrer le consignateur. A la place, le contrôleur de processus du consignateur effectue une action basée sur la valeur de la propriété de délai maxRestart.	130
maxRestartDelay	Détermine le comportement du contrôleur de processus du consignateur lorsque le taux de redémarrages du consignateur dépasse la valeur des propriétés d'intervalle maxRestart et maxRestart. Si vous indiquez une valeur inférieure ou égale à zéro, le contrôleur de processus du consignateur est arrêté. Si vous indiquez une valeur supérieure à zéro, il s'agit du nombre de secondes à attendre avant que les informations d'historique de redémarrage détenues par le contrôleur de processus du consignateur soient réinitialisées et que le consignateur soit redémarré.	-1
wmqfte.oracle.port	Port utilisé par le consignateur pour se connecter à l'instance Oracle . Ce port est également appelé programme d'écoute TNS.	1521
wmqfte.oracle.host	Hôte utilisé par le consignateur pour se connecter à l'instance Oracle .	système hôte local

Tableau 47. Propriétés de connexion pour le mode liaisons (suite)

Nom de la propriété	Description	Valeur par défaut
armELEMTYPE	Propriété facultative. Si le consignateur est configuré pour être redémarré par le gestionnaire de redémarrage automatique (ARM), définissez cette propriété sur la valeur du paramètre ARM ELEMTYPE spécifiée dans la règle ARM associée. Pour un consignateur, définissez ELEMTYPE sur SYSBFLG.	Non définie
armELEMENT	Propriété facultative. Si le consignateur est configuré pour être redémarré par le gestionnaire de redémarrage automatique (ARM), définissez cette propriété sur la valeur du paramètre ARM ELEMENT spécifiée dans la règle ARM associée. Vous pouvez définir la valeur ELEMENT pour qu'elle corresponde au nom du consignateur.	Non définie
loggerQMgrAuthenticationCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification de connexion MQ pour la connexion au gestionnaire de files d'attente de coordination du consignateur.	 Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir Création d'un fichier de données d'identification MFT  Pour plus d'informations sur l'emplacement et les droits d'accès de ce fichier, voir Configuration de MQMFTCredentials.xml .  Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir MFT et IBM MQ authentification de la connexion .
trace	<p>Propriété facultative. Spécification de trace lorsque le consignateur doit être exécuté avec la trace activée au démarrage du consignateur. La spécification de trace est une liste de classes séparée par des virgules, suivie du signe égal et d'un niveau de trace.</p> <p>Par exemple, <code>com.ibm.wmqfte.databaselogger,com.ibm.wmqfte.databaselogger.operation=all</code>.</p> <p>Vous pouvez spécifier plusieurs spécifications de trace dans une liste de valeurs séparées par deux-points. Par exemple : <code>com.ibm.wmqfte.databaselogger=moderate:com.ibm.wmqfte.databaselogger.operation=all</code></p>	Aucun
traceFiles	Propriété facultative. Nombre total de fichiers de trace à conserver. Cette valeur s'applique au contrôleur de processus d'un consignateur, ainsi qu'au consignateur lui-même.	5
traceSize	Propriété facultative. Taille maximale en Mo de chaque fichier de trace, avant que la trace passe au fichier suivant. Cette valeur s'applique au contrôleur de processus du consignateur et au consignateur lui-même.	20

Propriétés de connexion en mode client

V 9.0.4

Le tableau suivant présente les propriétés de consignateur supplémentaires requises pour la prise en charge de la connexion en mode client à un gestionnaire de files d'attente de consignateur.

Tableau 48. Propriétés de connexion pour le mode client

Nom de la propriété	Description	Valeur par défaut
wmqfte.queue.manager.host	Nom d'hôte ou adresse IP du gestionnaire de files d'attente du consignateur.	Aucune valeur par défaut
wmqfte.queue.manager.port	Port sur lequel le gestionnaire de files d'attente du consignateur est en mode écoute.	1414
wmqfte.queue.manager.channel	Nom du canal de connexion serveur sur le gestionnaire de files d'attente du consignateur.	SYSTEM.DEF.SVRCONN
wmqfte.Ssl.CipherSuite	Indique les aspects TLS de la façon dont le consignateur et le gestionnaire de files d'attente du consignateur échangent des données. La valeur de wmqfte.Ssl.CipherSuite est un nom CipherSuite . Le nom CipherSuite est mappé au nom CipherSpec utilisé sur le canal du gestionnaire de files d'attente du consignateur. Pour plus d'informations, voir CipherSuite et CipherSpec name mappings .	Aucune valeur par défaut
wmqfte.Ssl.PeerName	Indique un squelette de nom distinctif qui doit correspondre au nom fourni par le gestionnaire de files d'attente du consignateur. Le nom distinctif est utilisé pour vérifier le certificat d'identification présenté par le gestionnaire de files d'attente lors de la connexion.	Aucune valeur par défaut
wmqfte.Ssl.TrustStore	Indique l'emplacement des certificats auxquels le consignateur fait confiance. La valeur de wmqfte.Ssl.TrustStore est un chemin de fichier. Si le chemin d'accès au fichier est un chemin d'accès au fichier Windows , la barre oblique inversée (\) doit être mise en échappement avec une autre barre oblique inversée (\\). Notez que la valeur de cette propriété peut contenir des variables d'environnement.	Aucune valeur par défaut
wmqfte.Ssl.TrustStoreCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification wmqfte.Ssl.TrustStore . Notez que la valeur de cette propriété peut contenir des variables d'environnement.	Aucune valeur par défaut
wmqfte.Ssl.TrustStoreType	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être jks ou pkcs12.	jks
wmqfte.Ssl.KeyStore	Indique l'emplacement de la clé privée du consignateur. La valeur de wmqfte.Ssl.KeyStore est un chemin de fichier. Si le chemin d'accès au fichier est un chemin d'accès au fichier Windows , la barre oblique inversée (\) doit être mise en échappement avec une autre barre oblique inversée (\\). Notez que la valeur de cette propriété peut contenir des variables d'environnement.	Aucune valeur par défaut

Tableau 48. Propriétés de connexion pour le mode client (suite)

Nom de la propriété	Description	Valeur par défaut
wmqfte.Ssl.KeyStore.CredentialsFile	Chemin d'accès au fichier qui contient les données d'identification wmqfte.Ssl.KeyStore . Notez que la valeur de cette propriété peut contenir des variables d'environnement.	Aucune valeur par défaut
wmqfte.Ssl.KeyStoreType	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être jks ou pkcs12.	jks
wmqfte.Ssl.FipsRequired	Indique que vous souhaitez activer la prise en charge de FIPS au niveau du consignateur. La valeur de cette propriété peut être true ou false. Pour plus d'informations, voir «Prise en charge de FIPS dans MFT», à la page 688.	false

Référence associée

«Utilisation des variables d'environnement dans les propriétés MFT», à la page 547

Depuis la IBM WebSphere MQ 7.5, il est possible d'utiliser des variables d'environnement dans les propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction des changements d'environnement. Par exemple, l'utilisateur qui exécute le processus.

«Propriétés SSL pour MFT», à la page 622

Utilisez SSL ou TLS avec IBM MQ et Managed File Transfer pour empêcher les connexions non autorisées entre les agents et les gestionnaires de files d'attente et pour chiffrer le trafic des messages entre les agents et les gestionnaires de files d'attente.

Propriétés système Java pour MFT

Un certain nombre de propriétés de commande et d'agent Managed File Transfer doivent être définies en tant que propriétés système Java, car elles définissent la configuration de la fonction précoce qui ne peut pas utiliser le mécanisme des propriétés de commande ou d'agent.

Définissez les propriétés système et les autres options JVM pour la machine virtuelle Java qui doit exécuter les commandes Managed File Transfer en définissant la variable d'environnement BFG_JVM_PROPERTIES. Par exemple, pour définir la propriété com.ibm.wmqfte.maxConsoleLineLength sur une plateforme de type UNIX, définissez la variable comme suit:

```
export BFG_JVM_PROPERTIES="-Dcom.ibm.wmqfte.maxConsoleLineLength=132"
```

Si vous exécutez un agent en tant que service Windows, vous pouvez modifier les propriétés système Java de l'agent en spécifiant le paramètre `-sj` dans la commande **fteModifyAgent**.

Tableau 49. JavaPropriétés système

Nom de la propriété	Description	Valeur
com.ibm.wmqfte.maxConsoleLineLength	Longueur maximale de la ligne pouvant être écrite sur la console. Les lignes qui dépassent cette longueur sont des retours à la ligne. Cette valeur est exprimée en octets (et non en caractères).	 La longueur par défaut de IBM i est de 132 octets.   Pour UNIX, Linux, Windows et z/OS, la longueur est illimitée.
com.ibm.wmqfte.daemon.windows.windowsServiceLogFilesm	(Windows uniquement.) Indique le nombre maximal de fichiers journaux de service Windows à conserver. Les fichiers journaux du service Windows sont créés dans les répertoires des journaux de l'agent et du consigneur de base de données si ces applications s'exécutent en tant que service Windows. Les fichiers journaux du service Windows sont nommés avec le préfixe <i>service</i> et contiennent des messages sur le démarrage et l'arrêt du service.	5

Concepts associés

«Options de configuration de MFT sur Multiplatforms», à la page 58

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.

«Conseils et astuces pour l'utilisation de MFT», à la page 279

Voici quelques suggestions pour vous aider à utiliser au mieux Managed File Transfer.

Propriétés SSL pour MFT

Utilisez SSL ou TLS avec IBM MQ et Managed File Transfer pour empêcher les connexions non autorisées entre les agents et les gestionnaires de files d'attente et pour chiffrer le trafic des messages entre les agents et les gestionnaires de files d'attente.

Les fichiers de propriétés MFT suivants incluent des propriétés SSL:

- Le fichier `agent.properties`
- Le fichier `coordination.properties`
- Le fichier `command.properties`
-  Le fichier `logger.properties`

Pour plus d'informations sur l'utilisation de SSL avec Managed File Transfer, voir «Configuration du chiffrement SSL ou TLS pour MFT», à la page 52.

Pour IBM WebSphere MQ 7.5 ou les versions ultérieures, il est possible d'utiliser des variables d'environnement dans certaines propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction des changements d'environnement, tels que l'utilisateur qui exécute le processus. Pour plus d'informations, voir la section «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.

Propriétés SSL du fichier agent.properties

Le fichier `agent.properties` d'un agent se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name`. Les propriétés qu'il contient incluent les propriétés SSL suivantes:

Tableau 50. Propriétés SSL du fichier `agent.properties`

Nom de la propriété	Description	Valeur par défaut
<code>agentSslCipherSpec</code>	<p>Indique le protocole, l'algorithme de hachage et l'algorithme de chiffrement utilisés, ainsi que le nombre de bits utilisés dans la clé de chiffrement, lorsque les données sont échangées entre l'agent et le gestionnaire de files d'attente de l'agent.</p> <p>La valeur de <code>agentSslCipherSpec</code> est un nom CipherSpec. Ce nom CipherSpec est identique au nom CipherSpec utilisé sur le canal du gestionnaire de files d'attente de l'agent. La liste des noms CipherSpec valides est incluse dans les CipherSpecs SSL/TLS CipherSpecs et CipherSuites dans les classes IBM MQ pour Java et SSL/TLS CipherSpecs et CipherSuites dans les classes IBM MQ pour JMS.</p> <p><code>agentSslCipherSpec</code> est similaire à <code>agentSslCipherSuite</code>. Si <code>agentSslCipherSuite</code> et <code>agentSslCipherSpec</code> sont spécifiés, la valeur de <code>agentSslCipherSpec</code> est utilisée.</p>	Aucun
<code>agentSslCipherSuite</code>	<p>Indique les aspects SSL de la façon dont l'agent et le gestionnaire de files d'attente de l'agent échangent des données.</p> <p>La valeur de <code>agentSslCipherSuite</code> est un nom CipherSuite. Le nom CipherSuite est mappé au nom CipherSpec utilisé sur le canal du gestionnaire de files d'attente de l'agent. Pour plus d'informations, voir <code>CipherSuite</code> et <code>CipherSpec name mappings</code>.</p> <p><code>agentSslCipherSuite</code> est similaire à <code>agentSslCipherSpec</code>. Si <code>agentSslCipherSuite</code> et <code>agentSslCipherSpec</code> sont spécifiés, la valeur de <code>agentSslCipherSpec</code> est utilisée.</p>	Aucun
<code>agentSslPeerName</code>	<p>Indique un squelette de nom distinctif qui doit correspondre au nom fourni par le gestionnaire de files d'attente d'agent. Le nom distinctif est utilisé pour vérifier le certificat d'identification présenté par le gestionnaire de files d'attente lors de la connexion.</p>	Aucun
<code>agentSslTrustStore</code>	<p>Indique l'emplacement des certificats auxquels l'agent fait confiance. La valeur de <code>agentSslTrustStore</code> est un chemin de fichier. S'il s'agit d'un chemin de fichier Windows, la barre oblique inversée (\) doit être mise en échappement (\\).</p> <p>Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.</p>	Aucun
<code>agentSslKeyStore</code>	<p>Indique l'emplacement de la clé privée de l'agent. La valeur de <code>agentSslKeyStore</code> est un chemin de fichier. S'il s'agit d'un chemin de fichier Windows, la barre oblique inversée (\) doit être mise en échappement (\\). Cette propriété est requise uniquement si le gestionnaire de files d'attente de l'agent requiert l'authentification du client.</p> <p>Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.</p>	Aucun

Tableau 50. Propriétés SSL du fichier <i>agent.properties</i> (suite)		
Nom de la propriété	Description	Valeur par défaut
agentSslFipsRequired	Indique que vous souhaitez activer la prise en charge FIPS au niveau de l'agent. La valeur de cette propriété peut être <code>true</code> ou <code>false</code> . Pour plus d'informations, voir la section «Prise en charge de FIPS dans MFT», à la page 688.	false
agentSslKeyStore	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être <code>jdk</code> ou <code>pkcs12</code> .	jdk
agentSslKeyStoreCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification <code>agentSslKeyStore</code> . Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.	La valeur par défaut de cette propriété est <code>%USERPROFILE%\MQMFTCredentials.xml</code> sur Windows et <code>\$HOME/MQMFTCredentials.xml</code> sur d'autres plateformes.
agentSslTrustStore	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être <code>jdk</code> ou <code>pkcs12</code> .	jdk
agentSslTrustStoreCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification <code>agentSslTrustStore</code> . Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.	La valeur par défaut de cette propriété est <code>%USERPROFILE%\MQMFTCredentials.xml</code> sur Windows et <code>\$HOME/MQMFTCredentials.xml</code> sur d'autres plateformes.

Propriétés SSL du fichier *coordination.properties*

Le fichier *coordination.properties* se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name`. Les propriétés qu'il contient incluent les propriétés SSL suivantes:

Tableau 51. Propriétés SSL du fichier <i>coordination.properties</i>		
Nom de la propriété	Description	Valeur par défaut
coordinationSslCipherSpec	Indique le protocole, l'algorithme de hachage et l'algorithme de chiffrement utilisés, ainsi que le nombre de bits utilisés dans la clé de chiffrement, lorsque les données sont échangées entre les commandes et le gestionnaire de files d'attente de coordination. La valeur de <code>coordinationSslCipherSpec</code> est un nom <code>CipherSpec</code> . Ce nom <code>CipherSpec</code> est identique au nom <code>CipherSpec</code> utilisé sur le canal du gestionnaire de files d'attente de coordination. La liste des noms <code>CipherSpec</code> valides est incluse dans les <code>CipherSpecs SSL/TLS CipherSpecs</code> et <code>CipherSuites</code> dans les classes IBM MQ pour Java et <code>SSL/TLS CipherSpecs</code> et <code>CipherSuites</code> dans les classes IBM MQ pour JMS. <code>coordinationSslCipherSpec</code> est similaire à <code>coordinationSslCipherSuite</code> . Si <code>coordinationSslCipherSuite</code> et <code>coordinationSslCipherSpec</code> sont spécifiés, la valeur de <code>coordinationSslCipherSpec</code> est utilisée.	Aucun

Tableau 51. Propriétés SSL du fichier *coordination.properties* (suite)

Nom de la propriété	Description	Valeur par défaut
coordinationSslCipherSuite	<p>Indique les aspects SSL de la façon dont les commandes et le gestionnaire de files d'attente de coordination échangent des données.</p> <p>La valeur de <code>coordinationSslCipherSuite</code> est un nom CipherSuite . Le nom CipherSuite est mappé au nom CipherSpec utilisé sur le canal du gestionnaire de files d'attente de l'agent. Pour plus d'informations, voir CipherSuite et CipherSpec name mappings.</p> <p><code>coordinationSslCipherSuite</code> est similaire à <code>coordinationSslCipherSpec</code>. Si <code>coordinationSslCipherSuite</code> et <code>coordinationSslCipherSpec</code> sont spécifiés, la valeur de <code>coordinationSslCipherSpec</code> est utilisée.</p>	Aucun
coordinationSslPeerName	<p>Indique un squelette de nom distinctif qui doit correspondre au nom fourni par le gestionnaire de files d'attente de coordination. Le nom distinctif est utilisé pour vérifier le certificat d'identification présenté par le gestionnaire de files d'attente de coordination lors de la connexion.</p>	Aucun
coordinationSslTrustStore	<p>Indique l'emplacement des certificats auxquels les commandes font confiance. La valeur de <code>coordinationSslTrustStore</code> est un chemin de fichier. S'il s'agit d'un chemin de fichier Windows , la barre oblique inversée (\) doit être mise en échappement (\\).</p> <p>Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.</p>	Aucun
coordinationSslTrustStore	<p>Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être <code>jks</code> ou <code>pkcs12</code>.</p>	jks
coordinationSslTrustStoreCredentialsFile	<p>Chemin d'accès au fichier qui contient les données d'identification <code>coordinationSslTrustStore</code> .</p> <p>Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.</p>	La valeur par défaut de cette propriété est <code>%USERPROFILE%\MQMFTCredentials.xml</code> sur Windows et <code>\$HOME/MQMFTCredentials.xml</code> sur d'autres plateformes.
coordinationSslKeyStore	<p>Indique l'emplacement de la clé privée des commandes. La valeur de <code>coordinationSslKeyStore</code> est un chemin de fichier. S'il s'agit d'un chemin de fichier Windows , la barre oblique inversée (\) doit être mise en échappement (\\). Cette propriété est requise uniquement si le gestionnaire de files d'attente de coordination requiert une authentification client.</p> <p>Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.</p>	Aucun
coordinationSslKeyStore	<p>Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être <code>jks</code> ou <code>pkcs12</code>.</p>	jks
coordinationSslKeyStoreCredentialsFile	<p>Chemin d'accès au fichier qui contient les données d'identification <code>coordinationSslKeyStore</code> .</p> <p>Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.</p>	La valeur par défaut de cette propriété est <code>%USERPROFILE%\MQMFTCredentials.xml</code> sur Windows et <code>\$HOME/MQMFTCredentials.xml</code> sur d'autres plateformes.

Tableau 51. Propriétés SSL du fichier `coordination.properties` (suite)

Nom de la propriété	Description	Valeur par défaut
<code>coordinationSslFipsRequired</code>	Indique que vous souhaitez activer la prise en charge FIPS au niveau du gestionnaire de file d'attente de coordination. La valeur de cette propriété peut être <code>true</code> ou <code>false</code> . Pour plus d'informations, voir la section «Prise en charge de FIPS dans MFT», à la page 688.	<code>false</code>

Propriétés SSL du fichier `command.properties`

Le fichier `command.properties` se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name`. Les propriétés qu'il contient incluent les propriétés SSL suivantes:

Tableau 52. Propriétés SSL du fichier `command.properties`

Nom de la propriété	Description	Valeur par défaut
<code>connectionSslCipherSpec</code>	Indique le protocole, l'algorithme de hachage et l'algorithme de chiffrement utilisés, ainsi que le nombre de bits utilisés dans la clé de chiffrement, lorsque les données sont échangées entre les commandes et le gestionnaire de files d'attente de commandes. La valeur de <code>connectionSslCipherSpec</code> est un nom <code>CipherSpec</code> . Ce nom de <code>CipherSpec</code> est identique au nom de <code>CipherSpec</code> utilisé sur le canal du gestionnaire de files d'attente de commandes. Une liste de noms de <code>CipherSpec</code> valides est incluse dans les <code>CipherSpecsSSL/TLS CipherSpecs</code> et les <code>CipherSuites</code> dans les classes IBM MQ pour Java et les <code>CipherSpecs CipherSpecs</code> et les <code>CipherSuites</code> dans les classes IBM MQ pour JMS. <code>connectionSslCipherSpec</code> est similaire à <code>connectionSslCipherSuite</code> . Si <code>connectionSslCipherSuite</code> et <code>connectionSslCipherSpec</code> sont spécifiés, la valeur de <code>connectionSslCipherSpec</code> est utilisée.	Aucun
<code>connectionSslCipherSuite</code>	Indique les aspects SSL de la façon dont les commandes et le gestionnaire de files d'attente de commandes échangent des données. La valeur de <code>connectionSslCipherSuite</code> est un nom <code>CipherSuite</code> . Le nom <code>CipherSuite</code> est mappé au nom <code>CipherSpec</code> utilisé sur le canal du gestionnaire de files d'attente de l'agent. Pour plus d'informations, voir <code>CipherSuite</code> et <code>CipherSpec name mappings</code> . <code>connectionSslCipherSuite</code> est similaire à <code>connectionSslCipherSpec</code> . Si <code>connectionSslCipherSuite</code> et <code>connectionSslCipherSpec</code> sont spécifiés, la valeur de <code>connectionSslCipherSpec</code> est utilisée.	Aucun
<code>connectionSslPeerName</code>	Indique un squelette de nom distinctif qui doit correspondre au nom fourni par le gestionnaire de files d'attente de commandes. Le nom distinctif est utilisé pour vérifier le certificat d'identification présenté par le gestionnaire de files d'attente de commandes lors de la connexion.	Aucun
<code>connectionSslTrustStore</code>	Indique l'emplacement des certificats auxquels les commandes font confiance. La valeur de <code>connectionSslTrustStore</code> est un chemin de fichier. S'il s'agit d'un chemin de fichier Windows, la barre oblique inversée (<code>\</code>) doit être mise en échappement (<code>\\</code>). Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.	Aucun
<code>connectionSslTrustStoreType</code>	Type de magasin de clés de confiance SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être <code>jks</code> ou <code>pkcs12</code> .	<code>jks</code>

Tableau 52. Propriétés SSL du fichier `command.properties` (suite)

Nom de la propriété	Description	Valeur par défaut
<code>connectionSslTrustStoreCredentialsFile</code>	Chemin d'accès au fichier qui contient les données d'identification <code>connectionSslTrustStore</code> . Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.	La valeur par défaut de cette propriété est <code>%USERPROFILE%\MQMFTcredentials.xml</code> sur Windows et <code>\$HOME/MQMFTcredentials.xml</code> sur d'autres plateformes.
<code>connectionSslKeyStore</code>	Indique l'emplacement de la clé privée des commandes. La valeur de <code>connectionSslKeyStore</code> est un chemin de fichier. S'il s'agit d'un chemin de fichier Windows , la barre oblique inversée (\) doit être mise en échappement (\\). Cette propriété est requise uniquement si le gestionnaire de files d'attente de commandes requiert l'authentification du client. Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.	Aucun
<code>connectionSslKeyStore</code>	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être <code>jks</code> ou <code>pkcs12</code> . Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.	<code>jks</code>
<code>connectionSslKeyStoreCredentialsFile</code>	Chemin d'accès au fichier qui contient les données d'identification <code>connectionSslKeyStore</code> . Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.	La valeur par défaut de cette propriété est <code>%USERPROFILE%\MQMFTcredentials.xml</code> sur Windows et <code>\$HOME/MQMFTcredentials.xml</code> sur d'autres plateformes.
<code>connectionSslFipsRequired</code>	Indique que vous souhaitez activer la prise en charge FIPS au niveau du gestionnaire de files d'attente de commandes. La valeur de cette propriété peut être <code>true</code> ou <code>false</code> . Pour plus d'informations, voir la section «Prise en charge de FIPS dans MFT», à la page 688.	<code>false</code>

Propriétés SSL du fichier `logger.properties`

V 9.0.4

Le fichier `logger.properties` se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name` . Les propriétés requises pour la prise en charge de la connexion en mode client à un gestionnaire de files d'attente de consignateur incluent les propriétés SSL suivantes:

Tableau 53. Propriétés SSL du fichier `logger.properties`

Nom de la propriété	Description	Valeur par défaut
<code>wmqfte.Ssl.CipherSuite</code>	Indique les aspects TLS de la façon dont le consignateur et le gestionnaire de files d'attente du consignateur échangent des données. La valeur de <code>wmqfte.Ssl.CipherSuite</code> est un nom <code>CipherSuite</code> . Le nom <code>CipherSuite</code> est mappé au nom <code>CipherSpec</code> utilisé sur le canal du gestionnaire de files d'attente du consignateur. Pour plus d'informations, voir CipherSuite et CipherSpec name mappings .	Aucune valeur par défaut

Tableau 53. Propriétés SSL du fichier `logger.properties` (suite)

Nom de la propriété	Description	Valeur par défaut
<code>wmqfte.Ssl.PeerName</code>	Indique un squelette de nom distinctif qui doit correspondre au nom fourni par le gestionnaire de files d'attente du consignateur. Le nom distinctif est utilisé pour vérifier le certificat d'identification présenté par le gestionnaire de files d'attente lors de la connexion.	Aucune valeur par défaut
<code>wmqfte.Ssl.TrustStore</code>	Indique l'emplacement des certificats auxquels le consignateur fait confiance. La valeur de <code>wmqfte.Ssl.TrustStore</code> est un chemin de fichier. Si le chemin d'accès au fichier est un chemin d'accès au fichier Windows, la barre oblique inversée (\) doit être mise en échappement avec une autre barre oblique inversée (\\). Notez que la valeur de cette propriété peut contenir des variables d'environnement.	Aucune valeur par défaut
<code>wmqfte.Ssl.TrustStoreCredentialsFile</code>	Chemin d'accès au fichier qui contient les données d'identification <code>wmqfte.Ssl.TrustStore</code> . Notez que la valeur de cette propriété peut contenir des variables d'environnement.	Aucune valeur par défaut
<code>wmqfte.Ssl.TrustStoreType</code>	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être <code>jks</code> ou <code>pkcs12</code> .	<code>jks</code>
<code>wmqfte.Ssl.KeyStore</code>	Indique l'emplacement de la clé privée du consignateur. La valeur de <code>wmqfte.Ssl.KeyStore</code> est un chemin de fichier. Si le chemin d'accès au fichier est un chemin d'accès au fichier Windows, la barre oblique inversée (\) doit être mise en échappement avec une autre barre oblique inversée (\\). Notez que la valeur de cette propriété peut contenir des variables d'environnement.	Aucune valeur par défaut
<code>wmqfte.Ssl.KeyStore.CredentialsFile</code>	Chemin d'accès au fichier qui contient les données d'identification <code>wmqfte.Ssl.KeyStore</code> . Notez que la valeur de cette propriété peut contenir des variables d'environnement.	Aucune valeur par défaut
<code>wmqfte.Ssl.KeyStoreType</code>	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être <code>jks</code> ou <code>pkcs12</code> .	<code>jks</code>
<code>wmqfte.Ssl.FipsRequired</code>	Indique que vous souhaitez activer la prise en charge de FIPS au niveau du consignateur. La valeur de cette propriété peut être <code>true</code> ou <code>false</code> . Pour plus d'informations, voir la section «Prise en charge de FIPS dans MFT», à la page 688.	<code>false</code>

CipherSpecs et suites de chiffrement SHA-2 pour MFT

Managed File Transfer prend en charge les CipherSpecs et CipherSuites SHA-2 .

Pour activer l'utilisation de SHA-2 CipherSpecs et CipherSuites dans IBM MQ 8.0, sur les connexions entre les agents et les gestionnaires de files d'attente IBM MQ, vous devez utiliser IBM JREs 6.0 SR13 FP2, 7.0 SR4 FP2 ou version ultérieure.

Pour activer l'utilisation de SHA-2 CipherSpecs et CipherSuites dans Managed File Transfer V8, pour la connexion à un serveur FTPS à l'aide du pont de protocole en mode FTPS, vous devez utiliser IBM JRES 6.0 SR13 FP2, 7.0 SR4 FP2 ou version ultérieure.

Pour plus d'informations sur les CipherSpecs et les CipherSuites disponibles pour les connexions entre les agents et les gestionnaires de files d'attente IBM MQ, voir [SSL CipherSpecs et CipherSuites](#).

Pour plus d'informations sur la configuration de CipherSpecs et de CipherSuites à utiliser avec l'agent de pont de protocole et les serveurs FTPS, voir «Prise en charge du serveur FTPS par le pont de protocole», à la page 685 et «Format du fichier de propriétés du pont de protocole», à la page 592.

Les derniers chiffrements détaillés dans Spécification de CipherSpecs dans MQ 8.0 ne sont pas pris en charge par la machine virtuelle Java IBM i. Par conséquent, la prise en charge de SHA-2 pour la plateforme IBM i couvre uniquement les chiffrements détaillés dans [Spécification de CipherSpecs dans MQ 7.5](#).

Si vous souhaitez vous conformer à la norme SP 800-131A, vous devez respecter les exigences suivantes:

- Vous devez utiliser FTPS, que vous avez configuré de manière appropriée ; SFTP n'est pas pris en charge.
- Le serveur distant doit envoyer des suites de chiffrement SP 800-131A-compliant uniquement.

Référence associée

«Propriétés SSL pour MFT», à la page 622

Utilisez SSL ou TLS avec IBM MQ et Managed File Transfer pour empêcher les connexions non autorisées entre les agents et les gestionnaires de files d'attente et pour chiffrer le trafic des messages entre les agents et les gestionnaires de files d'attente.

Fichiers de configuration du consignateur de fichiers MFT

Outre le fichier `logger.properties`, un consignateur de fichier autonome Managed File Transfer dispose également d'un fichier de configuration XML dans son répertoire de configuration. Ce fichier de configuration est appelé `FileLoggerFormat.xml` et il définit le format utilisé par le consignateur de fichiers pour écrire des messages dans le fichier journal. Le contenu de ce fichier doit être conforme au schéma XML défini dans le fichier `FileLoggerFormat.xsd`.

Format de journal par défaut du consignateur de fichiers autonome MFT

Définition du format de fichier journal par défaut pour le consignateur de fichier Managed File Transfer autonome.

```
<?xml version="1.0" encoding="UTF-8"?>
<logFormatDefinition xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="1.00" xsi:noNamespaceSchemaLocation="FileLoggerFormat.xsd">
  <messageTypes>
    <callCompleted>
      <format>
        <inserts>
          <insert type="user" width="19" ignoreNull="false">/transaction/action/@time</insert>
          <insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
          <insert type="system" width="6" ignoreNull="false">type</insert>
          <insert type="user" width="3" ignoreNull="false">/transaction/status/@resultCode</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/agent/@agent</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/agent/@QMgr</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/job/name</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/command/
@type</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/command/
@name</insert>
          <insert type="system" width="0" ignoreNull="true">callArguments</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/callResult/
@outcome</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/callResult/
result/error</insert>
        </inserts>
        <separator>;</separator>
      </format>
    </callCompleted>
    <callStarted>
      <format>
```

```

<inserts>
  <insert type="user" width="19" ignoreNull="false">/transaction/action/@time</insert>
  <insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
  <insert type="system" width="6" ignoreNull="false">type</insert>
  <insert type="user" width="0" ignoreNull="false">/transaction/agent/@agent</insert>
  <insert type="user" width="0" ignoreNull="false">/transaction/agent/@QMgr</insert>
  <insert type="user" width="0" ignoreNull="false">/transaction/job/name</insert>
  <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/command/
@type</insert>
  <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/command/
@name</insert>
  <insert type="system" width="0" ignoreNull="true">callArguments</insert>
</inserts>
</separator>;</separator>
</format>
</callStarted>
<monitorAction>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/monitorLog/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/monitorLog/@referenceId</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="false">/monitorLog/status/@resultCode</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/@monitorName</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@agent</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@QMgr</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/action</insert>
    </inserts>
    <separator>;</separator>
  </format>
</monitorAction>
<monitorCreate>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/monitorLog/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/monitorLog/@referenceId</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/@monitorName</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@agent</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@QMgr</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/action</insert>
    </inserts>
    <separator>;</separator>
  </format>
</monitorCreate>
<monitorFired>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/monitorLog/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/monitorLog/@referenceId</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="false">/monitorLog/status/@resultCode</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/@monitorName</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@agent</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@QMgr</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/action</insert>
      <insert type="user" width="48" ignoreNull="false">/monitorLog/references/taskRequest</insert>
    </inserts>
    <separator>;</separator>
  </format>
</monitorFired>
<notAuthorized>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/notAuthorized/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/notAuthorized/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="false">/notAuthorized/status/@resultCode</insert>
      <insert type="user" width="12" ignoreNull="false">/notAuthorized/action</insert>
      <insert type="user" width="12" ignoreNull="false">/notAuthorized/authority</insert>
      <insert type="user" width="0" ignoreNull="false">/notAuthorized/originator/userID</insert>
      <insert type="user" width="0" ignoreNull="false">/notAuthorized/status/supplement</insert>
    </inserts>
    <separator>;</separator>
  </format>
</notAuthorized>
<scheduleDelete>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>

```

```

        <insert type="user" width="3" ignoreNull="false">/schedulelog/status/@resultCode</insert>
        <insert type="user" width="0" ignoreNull="false">/schedulelog/sourceAgent/@agent</insert>
        <insert type="user" width="12" ignoreNull="false">/schedulelog/action</insert>
        <insert type="user" width="0" ignoreNull="false">/schedulelog/originator/userID</insert>
        <insert type="user" width="0" ignoreNull="true">/schedulelog/status/supplement</insert>
    </inserts>
    <separator>;</separator>
</format>
</scheduleDelete>
<scheduleExpire>
    <format>
        <inserts>
            <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
            <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
            <insert type="system" width="6" ignoreNull="false">type</insert>
            <insert type="user" width="3" ignoreNull="false">/schedulelog/status/@resultCode</insert>
            <insert type="user" width="0" ignoreNull="false">/schedulelog/sourceAgent/@agent</insert>
            <insert type="user" width="12" ignoreNull="false">/schedulelog/action</insert>
            <insert type="user" width="0" ignoreNull="false">/schedulelog/originator/userID</insert>
            <insert type="user" width="0" ignoreNull="true">/schedulelog/status/supplement</insert>
        </inserts>
        <separator>;</separator>
    </format>
</scheduleExpire>
<scheduleSkipped>
    <format>
        <inserts>
            <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
            <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
            <insert type="system" width="6" ignoreNull="false">type</insert>
            <insert type="user" width="3" ignoreNull="false">/schedulelog/status/@resultCode</insert>
            <insert type="user" width="0" ignoreNull="false">/schedulelog/sourceAgent/@agent</insert>
            <insert type="user" width="12" ignoreNull="false">/schedulelog/action</insert>
            <insert type="user" width="0" ignoreNull="false">/schedulelog/originator/userID</insert>
            <insert type="user" width="0" ignoreNull="true">/schedulelog/status/supplement</insert>
        </inserts>
        <separator>;</separator>
    </format>
</scheduleSkipped>
<scheduleSubmitInfo>
    <format>
        <inserts>
            <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
            <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
            <insert type="system" width="6" ignoreNull="false">type</insert>
            <insert type="user" width="3" ignoreNull="false">/schedulelog/status/@resultCode</insert>
            <insert type="user" width="0" ignoreNull="false">/schedulelog/sourceAgent/@agent</insert>
            <insert type="user" width="12" ignoreNull="false">/schedulelog/action</insert>
            <insert type="user" width="0" ignoreNull="false">/schedulelog/originator/userID</insert>
            <insert type="user" width="0" ignoreNull="true">/schedulelog/schedule/submit/</insert>
            <insert type="user" width="0" ignoreNull="true">/schedulelog/schedule/submit/@timezone</
insert>
            <insert type="user" width="3" ignoreNull="true">/schedulelog/schedule/repeat/frequency/<
insert>
            <insert type="user" width="12" ignoreNull="true">/schedulelog/schedule/repeat/frequency/
@interval</insert>
            <insert type="user" width="3" ignoreNull="true">/schedulelog/schedule/repeat/expireCount</
insert>
            <insert type="user" width="0" ignoreNull="true">/schedulelog/status/supplement</insert>
        </inserts>
        <separator>;</separator>
    </format>
</scheduleSubmitInfo>
<scheduleSubmitTransfer>
    <format>
        <inserts>
            <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
            <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
            <insert type="system" width="10" ignoreNull="false">type</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/sourceAgent/@agent |
/transaction/sourceWebUser/@webGatewayAgentName |
/transaction/sourceWebGateway/@webGatewayAgentName</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/sourceAgent/@QMgr |
/transaction/sourceWebUser/@webGatewayAgentQMgr |
/transaction/sourceWebGateway/@webGatewayAgentQMgr</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/destinationAgent/@agent |
/transaction/destinationWebUser/@webGatewayAgentName |
/transaction/destinationWebGateway/@webGatewayAgentName</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/destinationAgent/@QMgr |
/transaction/destinationWebUser/@webGatewayAgentQMgr |
/transaction/destinationWebGateway/@webGatewayAgentQMgr</insert>
        </inserts>

```

```

    <separator>;</separator>
  </format>
</scheduleSubmitTransfer>
<scheduleSubmitTransferSet>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
      <insert type="system" width="10" ignoreNull="false">type</insert>
      <insert type="user" width="0" ignoreNull="false">source/file | source/queue</insert>
      <insert type="user" width="5" ignoreNull="true">source/@type</insert>
      <insert type="user" width="6" ignoreNull="true">source/@disposition</insert>
      <insert type="user" width="0" ignoreNull="false">destination/file | destination/queue</
insert>
      <insert type="user" width="5" ignoreNull="true">destination/@type</insert>
      <insert type="user" width="9" ignoreNull="true">destination/@exist</insert>
    </inserts>
    <separator>;</separator>
  </format>
</scheduleSubmitTransferSet>
<transferStarted>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/transaction/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="true">/transaction/status/@resultCode</insert>
      <insert type="user" width="0" ignoreNull="false">/transaction/sourceAgent/@agent |
/transaction/sourceWebUser/@webGatewayAgentName |
/transaction/sourceWebGateway/@webGatewayAgentName</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/sourceAgent/@QMgr |
/transaction/sourceWebUser/@webGatewayAgentQMgr |
/transaction/sourceWebGateway/@webGatewayAgentQMgr</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/sourceAgent/@agentType |
/transaction/sourceWebUser/@webGatewayAgentType |
/transaction/sourceWebGateway/@webGatewayAgentType</insert>
      <insert type="user" width="0" ignoreNull="false">/transaction/destinationAgent/@agent |
/transaction/destinationWebUser/@webGatewayAgentName |
/transaction/destinationWebGateway/@webGatewayAgentName</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/destinationAgent/@QMgr |
/transaction/destinationWebUser/@webGatewayAgentQMgr |
/transaction/destinationWebGateway/@webGatewayAgentQMgr</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/originator/userID</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/job/name</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/scheduleLog/@ID</insert>
    </inserts>
    <separator>;</separator>
  </format>
</transferStarted>
<transferCancelled>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/transaction/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="true">/transaction/status/@resultCode</insert>
      <insert type="user" width="0" ignoreNull="false">/transaction/sourceAgent/@agent |
/transaction/sourceWebUser/@webGatewayAgentName |
/transaction/sourceWebGateway/@webGatewayAgentName</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/sourceAgent/@QMgr |
/transaction/sourceWebUser/@webGatewayAgentQMgr |
/transaction/sourceWebGateway/@webGatewayAgentQMgr</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/sourceAgent/@agentType |
/transaction/sourceWebUser/@webGatewayAgentType |
/transaction/sourceWebGateway/@webGatewayAgentType</insert>
      <insert type="user" width="0" ignoreNull="false">/transaction/destinationAgent/@agent |
/transaction/destinationWebUser/@webGatewayAgentName |
/transaction/destinationWebGateway/@webGatewayAgentName</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/destinationAgent/@QMgr |
/transaction/destinationWebUser/@webGatewayAgentQMgr |
/transaction/destinationWebGateway/@webGatewayAgentQMgr</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/destinationAgent/@agentType |
/transaction/destinationWebUser/@webGatewayAgentType |
/transaction/destinationWebGateway/@webGatewayAgentType</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/originator/userID</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/job/name</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/status/supplement</insert>
    </inserts>
    <separator>;</separator>
  </format>
</transferCancelled>
<transferComplete>

```

```

<format>
  <inserts>
    <insert type="user" width="19" ignoreNull="false">/transaction/action/@time</insert>
    <insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
    <insert type="system" width="6" ignoreNull="false">type</insert>
    <insert type="user" width="3" ignoreNull="true">/transaction/status/@resultCode</insert>
    <insert type="user" width="0" ignoreNull="false">/transaction/sourceAgent/@agent |
    /transaction/sourceWebUser/@webGatewayAgentName |
    /transaction/sourceWebGateway/@webGatewayAgentName</insert>
    <insert type="user" width="0" ignoreNull="true">/transaction/sourceAgent/@QMgr |
    /transaction/sourceWebUser/@webGatewayAgentQMgr |
    /transaction/sourceWebGateway/@webGatewayAgentQMgr</insert>
    <insert type="user" width="0" ignoreNull="true">/transaction/sourceAgent/@agentType |
    /transaction/sourceWebUser/@webGatewayAgentType |
    /transaction/sourceWebGateway/@webGatewayAgentType</insert>
    <insert type="user" width="0" ignoreNull="false">/transaction/destinationAgent/@agent |
    /transaction/destinationWebUser/@webGatewayAgentName |
    /transaction/destinationWebGateway/@webGatewayAgentName</insert>
    <insert type="user" width="0" ignoreNull="true">/transaction/destinationAgent/@QMgr |
    /transaction/destinationWebUser/@webGatewayAgentQMgr |
    /transaction/destinationWebGateway/@webGatewayAgentQMgr</insert>
    <insert type="user" width="0" ignoreNull="true">/transaction/destinationAgent/@agentType |
    /transaction/destinationWebUser/@webGatewayAgentType |
    /transaction/destinationWebGateway/@webGatewayAgentType</insert>
    <insert type="user" width="0" ignoreNull="true">/transaction/originator/userID</insert>
    <insert type="user" width="0" ignoreNull="true">/transaction/job/name</insert>
    <insert type="user" width="0" ignoreNull="true">/transaction/status/supplement</insert>
  </inserts>
  <separator>;</separator>
</format>
</transferComplete>
<transferDelete>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/transaction/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="true">/transaction/status/@resultCode</insert>
      <insert type="user" width="0" ignoreNull="false">/transaction/sourceAgent/@agent |
      /transaction/sourceWebUser/@webGatewayAgentName |
      /transaction/sourceWebGateway/@webGatewayAgentName</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/sourceAgent/@QMgr |
      /transaction/sourceWebUser/@webGatewayAgentQMgr |
      /transaction/sourceWebGateway/@webGatewayAgentQMgr</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/sourceAgent/@agentType |
      /transaction/sourceWebUser/@webGatewayAgentType |
      /transaction/sourceWebGateway/@webGatewayAgentType</insert>
      <insert type="user" width="0" ignoreNull="false">/transaction/destinationAgent/@agent |
      /transaction/destinationWebUser/@webGatewayAgentName |
      /transaction/destinationWebGateway/@webGatewayAgentName</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/destinationAgent/@QMgr |
      /transaction/destinationWebUser/@webGatewayAgentQMgr |
      /transaction/destinationWebGateway/@webGatewayAgentQMgr</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/destinationAgent/@agentType |
      /transaction/destinationWebUser/@webGatewayAgentType |
      /transaction/destinationWebGateway/@webGatewayAgentType</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/originator/userID</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/job/name</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/status/supplement</insert>
    </inserts>
    <separator>;</separator>
  </format>
</transferDelete>
<transferProgress>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/transaction/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="true">status/@resultCode</insert>
      <insert type="user" width="0" ignoreNull="false">source/file | source/queue</insert>
      <insert type="user" width="0" ignoreNull="false">source/file/@size | source/queue/@size</
insert>
      <insert type="user" width="5" ignoreNull="true">source/@type</insert>
      <insert type="user" width="6" ignoreNull="true">source/@disposition</insert>
      <insert type="user" width="0" ignoreNull="true">source/file/@alias | source/queue/@alias</
insert>
      <insert type="user" width="0" ignoreNull="true">source/file/@filesize | source/queue/
@filesize</insert>
      <insert type="user" width="0" ignoreNull="true">source/@correlationBoolean1</insert>
      <insert type="user" width="0" ignoreNull="true">source/@correlationNum1</insert>
      <insert type="user" width="0" ignoreNull="true">source/@correlationString1</insert>

```

```

insert> <insert type="user" width="0" ignoreNull="false">destination/file | destination/queue</
insert> <insert type="user" width="0" ignoreNull="false">destination/file/@size | destination/queue/
@size</insert>
<insert type="user" width="5" ignoreNull="true">destination/@type</insert>
<insert type="user" width="9" ignoreNull="true">destination/@exist</insert>
<insert type="user" width="0" ignoreNull="true">destination/file/@alias | destination/queue/
@alias</insert>
<insert type="user" width="0" ignoreNull="true">destination/file/@filesystem | destination/
queue/@filesystem</insert>
<insert type="user" width="0" ignoreNull="true">destination/file/@truncateRecords</insert>
<insert type="user" width="0" ignoreNull="true">destination/@correlationBoolean1</insert>
<insert type="user" width="0" ignoreNull="true">destination/@correlationNum1</insert>
<insert type="user" width="0" ignoreNull="true">destination/@correlationString1</insert>
<insert type="user" width="0" ignoreNull="true">status/supplement</insert>
</inserts>
<separator>;</separator>
</format>
</transferProgress>
</messageTypes>
</logFormatDefinition>

```

Référence associée

«Format du consignateur de fichiers MFT autonome», à la page 110

Le format des informations de message écrites par le consignateur de fichiers peut être défini dans le fichier FileLoggerFormat.xml.

«Format XSD du consignateur de fichier autonome», à la page 634

Schéma d'un format de fichier autonome.

Format XSD du consignateur de fichier autonome

Schéma d'un format de fichier autonome.

Schéma

```

<?xml version="1.0" encoding="UTF-8"?>
<!--
@start_non_restricted_prolog@
Version: %Z% %I% %W% %E% %U% [%H% %T%]

Licensed Materials - Property of IBM

5724-H72

Copyright IBM Corp. 2011, 2023. All Rights Reserved.

US Government Users Restricted Rights - Use, duplication or
disclosure restricted by GSA ADP Schedule Contract with
IBM Corp.
@end_non_restricted_prolog@
-->

<!--
This schema defines the format of the FileLoggerFormat XML file that contains the definition
of the format to use when logging FTE log messages to a file. When an XML file that conforms
to this schema is processed by a file logger it can contain definitions for one or more
message type(s) that define how log messages of those types are output to the file log.
-->

<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema">
<xsd:include schemaLocation="fteutils.xsd"/>

<!--
Defines the logFileDefinition and version number
<logFileDefinition version="1.00" ...
  <messageTypes>
    ...
  </messageTypes>
</logFileDefinition>
-->
<xsd:element name="logFileDefinition">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="messageTypes" type="messageTypesType" maxOccurs="1" minOccurs="1"/>
    </xsd:sequence>

```

```

        <xsd:attribute name="version"          type="versionType"          use="required"/>
    </xsd:complexType>
</xsd:element>

<!--
    Defines the set of accepted message types. The definition of individual message types
    is optional. If a particular types element is present but empty then no line will be
    output for messages of that type. If a particular types element is not present then
    the default format will be used to format messages of that type.
-->
<xsd:complexType name="messageTypesType">
    <xsd:sequence>
        <xsd:element name="callCompleted"          type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="callStarted"           type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="monitorAction"         type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="monitorCreate"        type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="monitorFired"         type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="notAuthorized"        type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="scheduleDelete"       type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="scheduleExpire"       type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="scheduleSkipped"      type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="scheduleSubmitInfo"   type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="scheduleSubmitTransfer" type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="scheduleSubmitTransferSet" type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="transferStarted"      type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="transferCancelled"    type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="transferComplete"     type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="transferDelete"       type="messageType"          maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="transferProgress"    type="messageType"          maxOccurs="1"
minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>

<!--
    Defines the content of a message type definition e.g.

    <callStarted>
    <format>
    ...
    </format>
    <callStarted>
-->
<xsd:complexType name="messageType">
    <xsd:sequence>
        <xsd:element name="format"          type="messageFormatType"          maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>

<!--
    Defines the content of a message format definition e.g.

    <format>
    <inserts>
    ...
    </inserts>
    <separator>;</separator>
    </format>
-->
<xsd:complexType name="messageFormatType">
    <xsd:sequence>
        <xsd:element name="inserts"          type="insertsType"          maxOccurs="1" minOccurs="1"/>
        <xsd:element name="separator"       type="separatorType"          maxOccurs="1" minOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>

<!--

```

Defines the content of the inserts element e.g.

```
<inserts>
  <insert ...>
  <insert ...>
  ...
</inserts>
-->
<xsd:complexType name="insertsType">
  <xsd:sequence>
    <xsd:element name="insert" type="insertType" maxOccurs="unbounded" minOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>

<!--
  Defines the content of an insert definition e.g.
-->
  <insert type="user" width="0" ignoreNull="true"/>/transaction/@ID</insert>
-->
<xsd:complexType name="insertType">
  <xsd:attribute name="type" type="insertTypeType" use="required"/>
  <xsd:attribute name="width" type="xsd:nonNegativeInteger" use="required"/>
  <xsd:attribute name="ignoreNull" type="xsd:boolean" use="required"/>
</xsd:complexType>

<!--
  Defines the accepted choices for the insert type attribute.
-->
<xsd:simpleType name="insertTypeType">
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="user"/>
    <xsd:enumeration value="system"/>
  </xsd:restriction>
</xsd:simpleType>
</xsd:schema>
```

Référence associée

«Format du consignateur de fichiers MFT autonome», à la page 110

Le format des informations de message écrites par le consignateur de fichiers peut être défini dans le fichier `FileLoggerFormat.xml`.

«Format de journal par défaut du consignateur de fichiers autonome MFT», à la page 629

Définition du format de fichier journal par défaut pour le consignateur de fichier Managed File Transfer autonome.

Rubrique SYSTEM.FTE

SYSTEM.FTE est une rubrique du gestionnaire de file d'attente de coordination utilisée par Managed File Transfer pour consigner des transferts et stocker des informations sur les agents, les moniteurs, les plannings et les modèles.

Structure de la rubrique

```
SYSTEM.FTE
  /Agents
    /agent_name
  /monitors
    /agent_name
  /Scheduler
    /agent_name
  /Templates
    /template_ID
  /Transfers
    /agent_name
    /transfer_ID
  /Log
    /agent_name
    /Monitors
    /schedule_ID
    /transfer_ID
```

SYSTEM.FTE/Agents/nom_agent

Cette rubrique contient une publication conservée qui décrit un agent sur votre réseau Managed File Transfer, et ses propriétés. Le message dans cette rubrique est mis à jour régulièrement avec le statut de l'agent. Pour plus d'informations, voir la section [«Format de message d'état de l'agent MFT»](#), à la page 757.

SYSTEM.FTE/monitors/nom_agent

Cette rubrique contient des publications conservées qui décrivent les moniteurs de ressources associés à l'agent *nom_agent*. Le fichier XML de la publication conservée est conforme au schéma `MonitorList.xsd`. Pour plus d'informations, voir la section [«Format des messages de la liste de surveillance MFT»](#), à la page 759.

SYSTEM.FTE/Scheduler/nom_agent

Cette rubrique contient une publication conservée qui décrit toutes les planifications actives qui sont associées à l'agent *nom_agent*. Le fichier XML de la publication conservée est conforme au schéma `ScheduleList.xsd`. Pour plus d'informations, voir la section [«Format de message de la liste de planification MFT»](#), à la page 764.

SYSTEM.FTE/Templates

Cette rubrique contient des publications conservées qui décrivent tous les modèles qui sont définis dans votre topologie Managed File Transfer.

- La publication associée à chaque modèle est publiée dans une sous-rubrique nommée `SYSTEM.FTE/Templates/template_ID`.

Pour un exemple du contenu de cette publication conservée, voir [«Exemple de message XML de modèle MFT»](#), à la page 768.

SYSTEM.FTE/Transfers/nom_agent

Cette rubrique contient des publications qui décrivent le statut des transferts depuis l'agent *nom_agent*. Les publications associées à chaque transfert sont publiées dans une sous-rubrique portant le nom `SYSTEM.FTE/Transfers/agent_name/transfer_ID`. Ces publications sont utilisées par le plug-in IBM MQ Explorer pour fournir des informations sur les transferts individuels. Le fichier XML de la publication est conforme au schéma `TransferStatus.xsd`. Pour plus d'informations, voir la section [«Format des messages d'état de transfert de fichiers»](#), à la page 768.

SYSTEM.FTE/Log/nom_agent

Cette rubrique contient des publications qui consignent les informations sur les transferts, les moniteurs et les planifications provenant de l'agent *nom_agent*. Ces publications peuvent être consignées par le consigneur de base de données pour fournir des enregistrements d'audit des événements qui surviennent sur votre réseau Managed File Transfer.

- Les publications associées à chaque transfert sont publiées dans une sous-rubrique portant le nom `SYSTEM.FTE/Log/agent_name/transfer_ID` et le code XML de la publication est conforme au schéma `TransferLog.xsd`. Pour plus d'informations, voir la section [«Formats des messages du journal de transfert de fichiers»](#), à la page 771.
- Les publications associées à chaque transfert planifié sont publiées dans une sous-rubrique portant le nom `SYSTEM.FTE/Log/agent_name/schedule_ID` et le code XML de la publication est conforme au schéma `ScheduleLog.xsd`. Pour plus d'informations, voir la section [«Formats de message du journal de transfert de fichiers planifiés»](#), à la page 793.
- Les publications associées à chaque moniteur sont publiées dans une sous-rubrique nommée `SYSTEM.FTE/Log/agent_name/Monitors/monitor_name/monitor_ID` et le code XML de la publication est conforme au schéma `MonitorLog.xsd`. Pour plus d'informations, voir la section [«Format des messages du journal du moniteur MFT»](#), à la page 798.

Paramètres de file d'attente de l'agent MFT

Les scripts de commande MQSC générés par la commande **fteCreateAgent** créent les files d'attente d'agent avec les paramètres définis sur les valeurs suivantes. Si vous n'utilisez pas les scripts MQSC fournis pour créer les files d'attente, mais que vous les créez manuellement, veuillez à définir les paramètres suivants sur les valeurs indiquées.

Files d'attente d'opérations d'agent

Les files d'attente d'opérations de l'agent portent les noms suivants:

- SYSTEME SYSTEM.FTE.COMMAND.*nom_agent*
- SYSTEME SYSTEM.FTE.DATA.*nom_agent*
- SYSTEME SYSTEM.FTE.EVENT.*nom_agent*
- SYSTEME SYSTEM.FTE.REPLY.*nom_agent*
- SYSTEME SYSTEM.FTE.STATE.*nom_agent*

Tableau 54. Paramètres de la file d'attente des opérations de

Paramètre	Valeur (le cas échéant)
DEFPRTY	0
DEFSOPT	PARTAGE
GET	Activée
MAXDEPTH	5 000
MAXMSGL	4194304
MSGDLVSQ	PRIORITY
PUT	Activée
RETINTVL	999999999
PARTAGER	
Pas de déclencheur	
SYNTAXE	NORMALE
REPLACE	

Files d'attente d'autorisations d'agent

Les files d'attente de droits de l'agent portent les noms suivants:

- SYSTEM.FTE.AUTHADM1.*nom_agent*
- SYSTEME SYSTEM.FTE.AUTHAGT1.*nom_agent*
- SYSTEM.FTE.AUTHMON1.*nom_agent*
- SYSTEM.FTE.AUTHOPS1.*nom_agent*
- SYSTEM.FTE.AUTHSCH1.*nom_agent*
- SYSTEM.FTE.AUTHTRN1.*nom_agent*

Tableau 55. Paramètres de file d'attente des droits d'accès

Paramètre	Valeur (le cas échéant)
DEFPRTY	0
DEFSOPT	PARTAGE
GET	Activée
MAXDEPTH	0
MAXMSGL	0
MSGDLVSQ	PRIORITY

Tableau 55. Paramètres de file d'attente des droits d'accès (suite)

Paramètre	Valeur (le cas échéant)
PUT	Activée
RETINTVL	999999999
PARTAGER	
Pas de déclencheur	
SYNTAXE	NORMALE
REPLACE	

Référence associée

«Agent fteCreate(créer un agent MFT)», à la page 399

La commande **fteCreateAgent** crée un Managed File Transfer Agent et sa configuration associée.

Les files d'attente système MFT et la rubrique système

Managed File Transfer possède plusieurs files d'attente système et une rubrique système à usage interne uniquement.

Toutes les files d'attente dont le nom commence par SYSTEM.FTE sont des files d'attente système internes pour Managed File Transfer (MFT). Ne supprimez pas ces files d'attente car cela empêche IBM MQ MFT de fonctionner correctement. [Tableau 56, à la page 639](#) indique le type de message dans chaque file d'attente:

Tableau 56. Noms de file d'attente, type et utilisation

Nom de la file d'attente	Type de file d'attente	Utilisation
SYSTEM.FTE.AUTHAGT1.agent_name	Droits d'accès	File d'attente pour la configuration des droits d'envoi et de réception des demandes de transfert.
SYSTEM.FTE.AUTHTRN1.agent_name	Droits d'accès	File d'attente pour la configuration des droits de démarrage et d'annulation des transferts gérés. Permet également de démarrer des appels gérés.
SYSTEM.FTE.AUTHMON1.agent_name	Droits d'accès	File d'attente pour la configuration des droits permettant à un utilisateur de créer ou de supprimer des moniteurs de ressources créés par le même utilisateur.
SYSTEM.FTE.AUTHOPS1.agent_name	Droits d'accès	File d'attente pour la configuration des droits de suppression des moniteurs de ressources et des transferts planifiés créés par un autre utilisateur.

Tableau 56. Noms de file d'attente, type et utilisation (suite)

Nom de la file d'attente	Type de file d'attente	Utilisation
SYSTEM.FTE.AUTHSCH1.agent_name	Droits d'accès	File d'attente pour la configuration des droits permettant de créer ou de supprimer des transferts planifiés créés par le même utilisateur.
SYSTEM.FTE.AUTHADM1.agent_name	Droits d'accès	File d'attente pour la configuration des droits d'arrêt de l'agent, à l'aide de l'option -m de la commande fteStopAgent .
SYSTEM.FTE.COMMAND.agent_name	Opération	File d'attente pour l'envoi de demandes de commande à un agent.
SYSTEM.FTE.DATA.agent_name	Opération	File d'attente utilisée par un agent de destination pour le stockage des données envoyées par un agent source.
SYSTEM.FTE.REPLY.agent_name	Opération	File d'attente de réception des réponses d'un agent de destination.
SYSTEM.FTE.STATE.agent_name	Opération	File d'attente pour le maintien du statut d'une demande de transfert.
SYSTEM.FTE.EVENT.agent_name	Opération	File d'attente pour le maintien de l'historique du moniteur de ressources.

Si un agent participe à des transferts de message à fichier ou de fichier à message, il se peut que la définition de la file d'attente SYSTEM.FTE.STATE.agent_name doive être modifiée pour permettre ces transferts gérés. Pour plus d'informations à ce sujet, voir [Conseils pour la définition des attributs MQ et des propriétés MFT associées à la taille des messages](#).



Avertissement : Vous ne devez pas modifier les définitions des autres files d'attente système.

De plus, ne modifiez pas et ne supprimez pas SYSTEM.FTE, car cette rubrique est également destinée à un usage interne uniquement.

Files d'attente temporaires

Managed File Transfer crée des files d'attente temporaires à différentes fins. Le nom de chaque file d'attente commence par WMQFTE. par défaut. (Le point fait partie du préfixe par défaut.) Pour changer ce préfixe, vous pouvez utiliser la propriété **dynamicQueuePrefix** dans le fichier `command.properties` ou le fichier `coordination.properties`, ou dans les deux. La propriété dans le fichier `command.properties` permet de définir le préfixe des files d'attente temporaires qui sont créées pour les réponses aux commandes requérant une réponse de l'agent. La propriété dans le fichier `coordination.properties` permet de définir le préfixe des files d'attente temporaires qui sont créées à d'autres fins, comme `WMQFTE.FTE.TIMECHCK.QUEUE`, où `WMQFTE` est la valeur définie par la propriété **dynamicQueuePrefix**.

Référence associée

«Restriction des droits utilisateur sur les actions de l'agent MFT», à la page 368

En plus d'utiliser des groupes pour gérer l'accès aux ressources, vous pouvez activer un niveau supplémentaire de sécurité afin de restreindre les actions d'agent Managed File Transfer qu'un utilisateur peut effectuer. Accordez des autorisations à un utilisateur dans une file d'attente d'autorisations d'agent pour permettre à l'utilisateur d'effectuer des actions d'agent spécifiques.

Conventions de dénomination des objets MFT

Utilisez les conventions de dénomination suivantes pour vos objets Managed File Transfer :

- Noms d'agent et de consignateur:
 - Peut comporter jusqu'à 28 caractères et n'est pas sensible à la casse.
 - Les caractères entrés en minuscules ou en casse mixte sont convertis en majuscules
 - Doit être conforme aux conventions de dénomination d'objet IBM MQ standard.
- Ces conventions sont détaillées comme suit: [Règles de dénomination des objets IBM MQ](#).
- En plus des conventions de dénomination des objets IBM MQ , les règles suivantes s'appliquent:
 - Le caractère barre oblique (/) ne peut pas être utilisé dans les noms d'agent ou de consignateur
 - Le caractère pourcentage (%) ne peut pas être utilisé dans les noms d'agent ou de consignateur.
- Les noms des propriétés dans les fichiers de propriétés sont sensibles à la casse.
- Les noms de gestionnaire de files d'attente sont sensibles à la casse.
- Les noms de fichier sont sensibles à la casse pour certaines plateformes.
- Noms de modèle de transfert et de moniteur de ressources:
 - Ne sont pas sensibles à la casse
 - Les caractères entrés en minuscules ou en casse mixte sont convertis en majuscules
 - Ne doit pas contenir d'astérisque (*), de pourcentage (%) ou de point d'interrogation (?)
- Les noms de serveur de fichiers de protocole doivent:
 - Être d'une longueur minimale de 2 caractères, mais il n'y a pas de limite de longueur maximale
 - Ne sont pas sensibles à la casse
 - Doit être conforme aux conventions de dénomination d'objet IBM MQ standard.

Ces conventions sont détaillées comme suit: [Règles de dénomination des objets IBM MQ](#).

Fichiers du système de fichiers intégré IBM i (IFS)

 Les noms de fichier du système de fichiers intégré ne peuvent contenir aucun des caractères suivants:

- Barre oblique inversée (\)
- la barre oblique (/)
- Deux-points (:)
- Astérisque (*)
- Point d'interrogation (?)
- Guillemets (")
- Symbole inférieur à (<)
- Symbole supérieur à (>)
- Barre verticale (|)

Si vous tentez de transférer des fichiers dont le nom contient l'un de ces caractères vers un système IBM i IFS, le transfert de ces fichiers échoue.

Noms de fichier

z/OS Les fichiers sont soumis à des restrictions de dénomination qui affectent la longueur maximale du nom et les caractères disponibles que vous pouvez utiliser pour les noms de fichier. Les noms de membre de fichier PDS peuvent comporter jusqu'à huit caractères et ne peuvent pas contenir de point (.). Lorsque vous effectuez un transfert vers un fichier, vous devez spécifier explicitement le nom, ce qui signifie que ces restrictions de dénomination ne posent pas de problème. Toutefois, lorsque vous effectuez un transfert depuis des fichiers vers des membres PDS, le chemin d'accès au fichier peut ne pas être mappé vers un nom de membre PDS. Lorsque vous effectuez un transfert vers un fichier PDS, chaque fichier source devient un membre PDS et chaque nom de membre est généré à partir du nom de la source.

Les noms de membre PDS sont des noms z/OS non qualifiés et sont définis par l'expression régulière suivante:

```
[a-zA-Z$#@] [a-zA-Z0-9$#@] {0-7}
```

Le schéma suivant permet de convertir un nom de fichier source ou de fichier source en nom de membre PDS valide. Les considérations sont appliquées dans l'ordre indiqué:

1. Seuls les caractères du nom après la dernière barre oblique (/), la dernière barre oblique inversée (\) ou le dernier signe deux-points (:) sont utilisés. C'est-à-dire que seule la partie nom d'un chemin de fichier est utilisée.
2. Pour les fichiers source (qui ne sont pas des ensembles de données ou des membres PDS), les caractères après et incluant le dernier point (.) sont ignorés.
3. Pour tout nom de plus de huit caractères, seuls les huit derniers caractères sont utilisés.
4. Les caractères point sont remplacés par des caractères arobase (@).
5. Les caractères non valides sont remplacés par des caractères arobase (@).
6. Si la conversion ne génère aucun caractère, le nom du membre PDS est @.

Référence d'administration MFT

Valeurs de statut de l'agent MFT

Les commandes **ftelistagents** et **fteshowagentdetails** produisent des informations sur le statut de l'agent. Il existe plusieurs valeurs possibles pour ce statut.

ACTIF

L'agent est en cours d'exécution et envoie ou reçoit des fichiers. L'agent publie son statut à intervalles réguliers. La dernière mise à jour a été reçue dans le délai prévu.

PRET

L'agent est en cours d'exécution, mais il n'envoie ni ne reçoit de fichiers. L'agent publie son statut à intervalles réguliers. La dernière mise à jour a été reçue dans le délai prévu.

EN COURS DE DEMARRAGE

L'agent est en cours de démarrage, mais n'est pas encore prêt à réaliser des transferts.

IMPOSSIBLE A ATTEINDRE

Les mises à jour du statut de l'agent n'ont pas été reçues aux intervalles prévus. Il se peut que l'agent se soit arrêté en raison d'une erreur, qu'il ait été arrêté brutalement ou qu'il soit en cours d'exécution mais qu'il rencontre des problèmes de communication. Pour plus d'informations, voir la section «[Que faire si un agent est affiché comme étant à l'état UNKNOWN \(inconnu\)](#)», à la page 311.

ARRETE

L'agent a été arrêté. Son arrêt a eu lieu de façon contrôlée.

Arrêt inattendu

L'agent s'est arrêté de manière inattendue. L'agent sera automatiquement redémarré, sauf s'il y a eu plus de `maxRestartCount` redémarrages au cours de la période `maxRestartInterval` et que la

valeur `maxRestartDelay` est inférieure ou égale à 0. Pour plus d'informations sur ces propriétés, voir «Le fichier `MFT.agent.properties`», à la page 564.

INFORMATIONS SUR LE NOEUD

La version de l'agent peut être IBM WebSphere MQ File Transfer Edition 7.0.2 ou une version antérieure. L'agent ne publie pas les mises à jour dans un format que cette commande peut traiter.

INCONNU

Le statut de l'agent ne peut pas être déterminé. Il se peut qu'il ait publié un statut qui n'est pas reconnu par cet outil. Si vous disposez de versions de produits mixtes sur votre réseau, la mise à niveau de la version d'installation de cet outil peut résoudre ce problème.

V 9.0.2 Dans IBM MQ 9.0.2, lorsque vous exécutez des commandes ou que vous examinez la liste des agents qui se connectent à un gestionnaire de coordination et leurs propriétés individuelles, vous pouvez voir une nouvelle valeur **Status Age** pour l'agent qui indique l'ancienneté de leur dernier statut signalé. Pour plus d'informations, voir la section «Que faire si un agent est affiché comme étant à l'état UNKNOWN (inconnu)», à la page 311.

PROBLEME

Il se peut que le gestionnaire de commandes de l'agent ne fonctionne pas. L'agent publie des messages d'état, mais ces derniers sont obsolètes.

Référence associée

«Agents `fteList`: répertorie les agents MFT pour un gestionnaire de files d'attente de coordination», à la page 484

La commande **`fteListAgents`** permet de répertorier tous les agents Managed File Transfer enregistrés auprès d'un gestionnaire de files d'attente de coordination particulier.

«`fteShowAgentDetails`», à la page 530

La commande **`fteShowAgentDetails`** permet d'afficher les détails d'un Managed File Transfer Agent particulier. Il s'agit des détails stockés par le gestionnaire de files d'attente de coordination Managed File Transfer de l'agent.

«Que faire si vous pensez que votre transfert de fichiers est bloqué», à la page 321

Sur un système très chargé ou en cas de problèmes de réseau entre les agents source et cible, les transferts peuvent parfois sembler bloqués dans un état de file d'attente ou de récupération. Un certain nombre de facteurs peuvent être à l'origine de cette situation.

«Que faire si un agent est affiché comme étant à l'état UNKNOWN (inconnu)», à la page 311

Votre agent est en cours d'exécution et répond correctement à la commande **`ftePingAgent`**, et les éléments sont transférés normalement. Toutefois, les commandes **`fteListAgents`** et **`fteShowAgentDetails`**, ainsi que le plug-in IBM MQ Explorer Managed File Transfer, indiquent que l'agent est à l'état UNKNOWN.

Multi Présentation du contrôleur de processus MFT

Le contrôleur de processus IBM MQ Managed File Transfer (MFT) est chargé de démarrer un agent MFT et de redémarrer ce processus s'il se termine pour une raison quelconque. Il existe un contrôleur de processus pour chaque processus d'agent.

Remarque : Le contrôleur de processus est applicable à IBM MQ for Multiplatforms uniquement.

z/OS Sous IBM MQ for z/OS, le processus d'agent est redémarré par le gestionnaire de redémarrage automatique (ARM). Pour plus d'informations à ce sujet, voir [Configuration de MFT pour z/OS Automatic Restart Manager \(ARM\)](#)

Fonctionnement du contrôleur de processus

Lorsque la commande **`fteStartAgent`** est exécutée, elle démarre une instance du contrôleur de processus pour cet agent et le contrôleur de processus démarre ensuite le processus d'agent.

Lorsque la commande **fteStopAgent** est exécutée, elle se connecte au contrôleur de processus de cet agent et lui envoie une demande d'arrêt. Le contrôleur de processus reçoit la demande, arrête le processus d'agent, puis s'arrête.

Le contrôleur de processus surveille le processus d'agent. Si le processus d'agent s'arrête de manière inattendue, le contrôleur de processus le redémarre.

Par défaut, si un processus d'agent s'arrête cinq fois dans un délai de deux minutes, le contrôleur de processus s'arrête et ne tente pas de redémarrer l'agent à nouveau. Dans ce cas, vous devez redémarrer l'agent manuellement à l'aide de la commande **fteStartAgent**.

Vous pouvez modifier ce comportement en modifiant les propriétés d'agent suivantes:

- **maxRestartCount**
- **maxRestartDelay**
- **maxRestartInterval**

Si vous avez configuré un agent pour qu'il se connecte à son gestionnaire de files d'attente d'agent à l'aide du transport BINDINGS, le contrôleur de processus crée une connexion à ce gestionnaire de files d'attente lorsqu'il démarre. Le contrôleur de processus surveille ensuite cette connexion.

Si la connexion est interrompue car le gestionnaire de files d'attente est devenu indisponible, le contrôleur de processus arrête l'agent, puis tente de rétablir la connexion à intervalles réguliers.

Le délai entre les tentatives de reconnexion est déterminé par la propriété d'agent **agentQMgrRetryInterval**. Une fois que le gestionnaire de files d'attente est à nouveau disponible et que le contrôleur de processus a pu s'y connecter, le contrôleur de processus redémarre le processus d'agent.

Remarque : Lorsqu'un agent est configuré pour se connecter à son gestionnaire de files d'attente d'agent à l'aide du transport CLIENT, le processus d'agent reste actif s'il est déconnecté du gestionnaire de files d'attente. Dans cette situation, le processus d'agent tente de se reconnecter à intervalles réguliers.

Pour plus d'informations sur les quatre propriétés mentionnées dans cette section, voir la section [Propriétés avancées de l'agent: Contrôleur de processus](#) de la rubrique *Fichier MFT agent.properties*.

Fichiers journaux du contrôleur de processus

Le contrôleur de processus écrit des messages d'information dans son journal des événements. Il s'agit d'un fichier appelé pceventN.log, où N est un nombre, qui se trouve dans le répertoire suivant:
MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name/logs/

La taille de chaque fichier journal des événements du contrôleur de processus et le nombre de fichiers d'historique sont déterminés par les propriétés d'agent **outputLogSize** et **outputLogFiles**.

Pour plus d'informations sur les propriétés mentionnées dans cette section, voir la section [Propriétés avancées de l'agent: traçage et journalisation](#) de la rubrique *Fichier MFT agent.properties*.

Remarque : Ces propriétés sont également utilisées pour déterminer la taille et le nombre de fichiers journaux de l'agent (appelés outputN.log) ainsi que les fichiers journaux du contrôleur de processus.

Les messages consignés dans le journal des événements du contrôleur de processus incluent l'identificateur de processus du contrôleur de processus et l'identificateur de processus du processus de l'agent. Voici quelques exemples de ces messages:

```
[21/06/2022 16:17:40.000 GMT Daylight Time] 00000000000049e0
ProcessContro I BFGPC0003I: IBM MQ Managed File Transfer process controller started.
Log files located at: C:\ProgramData\IBM\MQ\mqft\logs\QM1\agents\AGENT1.
```

```
[21/06/2022 16:17:55.000 GMT Daylight Time] 00000000000049e0
ProcessContro I BFGPC0007I: IBM MQ Managed File Transfer process controller with process
identifier 18736 started AGENT1@QM1 with process identifier 1748.
```

```
[21/06/2022 16:19:20.000 GMT Daylight Time] 00000000000049e0
ProcessContro I BFGPC0027W: Process has ended with return code 1 and will be
restarted to attempt to recover the problem.
```

```
[21/06/2022 16:19:20.000 GMT Daylight Time] 00000000000049e0
ProcessContro I BFGPC0007I: IBM MQ Managed File Transfer process controller with process
identifieur 18736 started AGENT1@QM1 with process identifieur 1304.
```

Ici, le contrôleur de processus associé à l'agent AGENT1 était en cours d'exécution avec l'identificateur de processus 18736.

Initialement, il a démarré le processus d'agent-l'identificateur de ce processus était 1748.

Peu de temps après le démarrage de l'agent, le contrôleur de processus a détecté qu'il s'était arrêté de manière inattendue et l'a donc redémarré. Après le redémarrage, l'identificateur de processus du processus d'agent est 1304.

Référence associée

«Codes d'exit du contrôleur de processus MFT», à la page 647

Si le contrôleur de processus Managed File Transfer s'arrête, un message BFGPC0004I est généré avec un code d'exit qui indique la raison pour laquelle le contrôleur de processus s'est arrêté.

Valeurs de statut du contrôleur de processus de l'agent MFT

La commande **fteShowAgentDetails** génère des informations sur le statut du contrôleur de processus de l'agent. Il existe plusieurs valeurs possibles pour ce statut.

EN ATTENTE

Le contrôleur de processus de l'agent attend que le gestionnaire de files d'attente soit disponible avant de démarrer l'agent.

DEMARRE

Le contrôleur de processus d'agent a démarré le processus d'agent.

ARRETE

Le contrôleur de processus d'agent a été arrêté, soit en raison d'une demande d'arrêt de l'agent, soit parce qu'il y a eu trop de redémarrages de processus d'agent dans l'intervalle de redémarrage.

Récupération en cours

Le processus d'agent s'est arrêté de manière inattendue et le contrôleur de processus va tenter de le redémarrer.

ISTOPPING

Le processus d'agent a reçu une demande d'arrêt immédiat. Une fois le processus d'agent arrêté, le contrôleur de processus s'arrête.

CARRET

Le processus d'agent a reçu une demande d'arrêt contrôlé. Une fois le processus d'agent arrêté, le contrôleur de processus s'arrête.

INCONNU

Le statut du contrôleur de processus de l'agent ne peut pas être déterminé. Il se peut que le contrôleur de processus de l'agent ne soit pas en cours d'exécution ou qu'il soit en cours d'exécution sur un système différent de celui sur lequel la commande **fteShowAgentDetails** a été exécutée.

Référence associée

«fteShowAgentDetails», à la page 530

La commande **fteShowAgentDetails** permet d'afficher les détails d'un Managed File Transfer Agent particulier. Il s'agit des détails stockés par le gestionnaire de files d'attente de coordination Managed File Transfer de l'agent.

Valeurs de statut du consignateur MFT

Les commandes **fteShowLoggerDetails** produisent des informations sur le statut du consignateur. Il existe plusieurs valeurs possibles pour ce statut.

ACTIF

Le consignateur est en cours d'exécution et envoie ou reçoit des fichiers. Le consignateur publie son statut à intervalles réguliers. La dernière mise à jour a été reçue dans le délai prévu.

PRET

Le consignateur est en cours d'exécution, mais il n'envoie ni ne reçoit de fichiers. Le consignateur publie son statut à intervalles réguliers. La dernière mise à jour a été reçue dans le délai prévu.

EN COURS DE DEMARRAGE

Le consignateur est en cours de démarrage, mais il n'est pas encore prêt à effectuer des transferts.

IMPOSSIBLE A ATTEINDRE

Les mises à jour de statut du consignateur n'ont pas été reçues aux intervalles de temps prévus. Il se peut que le consignateur se soit arrêté en raison d'une erreur, qu'il ait été arrêté brusquement ou qu'il soit en cours d'exécution mais qu'il ait rencontré des problèmes de communication.

ARRETE

Le consignateur a été arrêté. Son arrêt a eu lieu de façon contrôlée.

Arrêt inattendu

Le consignateur s'est arrêté de manière inattendue. Le consignateur sera automatiquement redémarré, sauf si le nombre de redémarrages de `maxRestartCount` est supérieur à la période `maxRestartInterval` et que la valeur `maxRestartDelay` est inférieure ou égale à 0. Pour plus d'informations sur ces propriétés, voir «[Propriétés de configuration du consignateur MFT](#)», à la page 120.

Pour la commande `fteShowLoggerDetails`, les détails de ce statut incluent un code de statut, qui correspond au code de sortie du processus de consignment. Voir "Codes de sortie de processus" pour la liste des codes de sortie connus.

INFORMATIONS SUR LE NOEUD

La version du consignateur peut être IBM WebSphere MQ File Transfer Edition 7.0.2 antérieure. Le consignateur ne publie pas les mises à jour dans un format que cette commande peut traiter.

INCONNU

Le statut du consignateur ne peut pas être déterminé. Il se peut qu'il ait publié un statut qui n'est pas reconnu par cet outil. Si vous disposez de versions de produits mixtes sur votre réseau, la mise à niveau de la version d'installation de cet outil peut résoudre ce problème.

PROBLEME

Il se peut que le gestionnaire de commandes du consignateur ne fonctionne pas. Le consignateur publie des messages d'état, mais ces messages d'état sont obsolètes.

Référence associée

«[fteShowLoggerDetails](#)», à la page 537

La commande `fteShowLoggerDetails` permet d'afficher les détails d'un consignateur Managed File Transfer particulier.

Valeurs de statut du contrôleur de processus du consignateur MFT

La commande `fteShowLoggerDetails` génère des informations sur le statut du contrôleur de processus du consignateur. Il existe plusieurs valeurs possibles pour ce statut.

EN ATTENTE

Le contrôleur de processus du consignateur attend que le gestionnaire de files d'attente soit disponible avant de démarrer le consignateur.

DEMARRE

Le contrôleur de processus de consignateur a démarré le processus de consignateur.

ARRETE

Le contrôleur de processus du consignateur a été arrêté, soit en raison d'une demande d'arrêt du consignateur, soit parce qu'il y a eu trop de redémarrages de processus du consignateur dans l'intervalle de redémarrage.

Récupération en cours

Le processus du consignateur s'est arrêté de manière inattendue et le contrôleur de processus va tenter de le redémarrer.

ISTOPPING

Le processus du consignateur a reçu une demande d'arrêt immédiat. Une fois le processus du consignateur arrêté, le contrôleur de processus s'arrête.

CARRET

Le processus du consignateur a reçu une demande d'arrêt contrôlé. Une fois le processus du consignateur arrêté, le contrôleur de processus s'arrête.

INCONNU

Le statut du contrôleur de processus du consignateur ne peut pas être déterminé. Il se peut que le contrôleur de processus du consignateur ne soit pas en cours d'exécution ou qu'il soit en cours d'exécution sur un système différent de celui sur lequel la commande `fteShowLoggerDetails` a été exécutée.

Référence associée

«`fteShowLoggerDetails`», à la page 537

La commande **`fteShowLoggerDetails`** permet d'afficher les détails d'un consignateur Managed File Transfer particulier.

Codes d'exit du contrôleur de processus MFT

Si le contrôleur de processus Managed File Transfer s'arrête, un message BFGPC0004I est généré avec un code d'exit qui indique la raison pour laquelle le contrôleur de processus s'est arrêté.

Le message suivant s'affiche pour indiquer que le contrôleur de processus s'est arrêté:

```
BFGPC0004I IBM MQ Managed File Transfer s'est arrêté avec le code d'exit code_raison.
```

où *code_raison* indique la raison pour laquelle le contrôleur de processus s'est arrêté.

Remarque : Les codes de sortie du contrôleur de processus reflètent principalement les codes de sortie du système d'exploitation standard, mais certains codes de sortie sont définis à des fins spécifiques et sont toujours accompagnés d'un message spécifique dans le fichier journal du contrôleur de processus.

Code raison	Description
RC_SUCCESS = 0	L'arrêt du contrôleur de processus a abouti.
RC_FAILURE = 1	Code retour d'échec du contrôleur de processus général (ne doit en général pas être renvoyé).
RC_EXIT = 2	La sortie du contrôleur de processus a été forcée (par exemple, un système de diagnostic a demandé l'arrêt du contrôleur de processus).
RC_ABEND = 70	Le contrôleur de processus a rencontré un problème irrémédiable et est en train de s'arrêter de force.
RC_QMUNAVAIL = 75	Le contrôleur de processus ne peut pas continuer car le gestionnaire de files d'attente du contrôleur de processus n'est pas disponible.
RC_CONFIG = 78	Le contrôleur de processus ne peut pas continuer en raison d'un problème lié aux données de configuration de démarrage.

Ces codes d'exit sont écrits dans `pceventX.log`, où *X* peut être n'importe quel nombre, par exemple le nom du fichier journal peut être `pcevent0.log`.

Information associée

Messages de diagnostic Managed File Transfer: BFGPC0001 - BFGPC9999

Instructions pour le transfert de fichiers

Selon le système d'exploitation depuis et vers lequel vous effectuez le transfert et selon que vous effectuez le transfert en mode binaire ou texte, il existe des instructions sur le comportement à attendre.

Pour plus d'informations, consultez les rubriques correspondantes.

Référence associée

[«Transfert de fichiers et de jeux de données entre z/OS et les systèmes répartis», à la page 649](#)

Vous pouvez transférer des fichiers et des types d'ensemble de données pris en charge entre z/OS et les systèmes de fichiers distribués à l'aide de Managed File Transfer. Examinez attentivement le comportement suivant, qui dépend du type de système depuis et vers lequel vous effectuez le transfert.

[«Transfert entre fichiers sous z/OS», à la page 651](#)

Vous pouvez effectuer un transfert entre des fichiers z/OS à l'aide de Managed File Transfer. Examinez attentivement le comportement suivant pour vous assurer que vos fichiers sont transférés correctement.

[«Transfert de fichiers vers et depuis des noeuds Connect:Direct», à la page 653](#)

Vous pouvez transférer des ensembles de données entre les agents Managed File Transfer et les noeuds IBM Sterling Connect:Direct à l'aide du pont Connect:Direct . Vous pouvez spécifier un fichier comme source de transfert, destination de transfert ou les deux.

[«Mappages entre les paramètres d'instruction de processus Connect:Direct et les clés BPXWDYN», à la page 655](#)

Lorsque vous soumettez une demande de transfert pour un fichier dont la source ou la destination est un noeud Connect:Direct , toutes les clés BPXWDYN prises en charge que vous fournissez sont converties dans un format accepté par les processus Connect:Direct .

[«Propriétés BPXWDYN que vous ne devez pas utiliser avec MFT», à la page 661](#)

Certaines options BPXWDYN ne doivent pas être spécifiées en cas d'utilisation de la commande **fteCreateTemplate**, de la commande **fteCreateTransfer** ou de la propriété **bpxwdynAllocAdditionalOptions** dans le fichier `agent.properties`.

[«Transfert de fichiers texte avec MFT», à la page 662](#)

Le transfert de fichiers texte implique la conversion de la page de codes d'un fichier d'une page de codes à une autre. Le transfert de fichier texte implique également la conversion de caractères CRLF (retour chariot-saut de ligne) entre les systèmes. Cette rubrique récapitule le comportement de transfert de fichier texte de Managed File Transfer.

[«Transfert de fichiers texte entre Connect:Direct et MFT», à la page 665](#)

Le transfert de texte implique la conversion de la page de codes d'un fichier d'une page de codes à une autre. Le transfert de fichiers texte implique également la conversion de caractères CRLF (saut de ligne de retour chariot) entre les systèmes. Cette rubrique récapitule le comportement du transfert de fichiers texte dans les transferts entre un Managed File Transfer Agent et un noeud Connect:Direct .

[«Transfert de fichiers vers ou depuis des agents de pont de protocole», à la page 665](#)

Vous pouvez transférer des fichiers vers et depuis un serveur de fichiers FTP ou SFTP en dehors de votre réseau Managed File Transfer à l'aide d'un agent de pont de protocole.

[«Transfert de fichiers vers ou depuis des systèmes IBM i», à la page 666](#)

Si vous transférez des fichiers vers ou depuis des systèmes IBM i à l'aide de Managed File Transfer en mode texte et que vous souhaitez convertir les données des fichiers, prenez en compte les informations de cette rubrique.

[«Transfert des fichiers sauvegarde situés dans QSYS.LIB sous IBM i», à la page 671](#)

Managed File Transfer prend en charge le transfert des fichiers sauvegarde situés dans QSYS.LIB entre deux systèmes IBM i . Tenez compte des informations suivantes lorsque vous demandez des transferts de fichiers sauvegarde.

[«Transfert de groupes de données de génération \(GDG\)», à la page 672](#)

Managed File Transfer prend en charge les groupes de données de génération (GDG) pour les fichiers source et de destination sur z/OS. Les noms d'ensemble de fichiers absolus et relatifs sont pris en charge. Lorsque vous écrivez dans une nouvelle génération, l'ensemble de fichiers de base doit exister.

[«Utilisation de caractères génériques avec MFT», à la page 673](#)

Vous pouvez utiliser des caractères génériques lorsque vous spécifiez des noms de fichier source et des chemins de fichier source pour les transferts de fichiers. Cela vous permet de sélectionner plusieurs fichiers simultanément.

z/OS **Transfert de fichiers et de jeux de données entre z/OS et les systèmes répartis**

Vous pouvez transférer des fichiers et des types d'ensemble de données pris en charge entre z/OS et les systèmes de fichiers distribués à l'aide de Managed File Transfer. Examinez attentivement le comportement suivant, qui dépend du type de système depuis et vers lequel vous effectuez le transfert.

Managed File Transfer prend en charge les groupes de données de génération (GDG) pour les fichiers source et de destination sur z/OS. Les noms d'ensemble de fichiers absolus et relatifs sont pris en charge. Lorsque vous écrivez dans une nouvelle génération, l'ensemble de fichiers de base doit exister.

Lorsque vous transférez un fichier ou un ensemble de données sur bande, tout ensemble de données existant déjà sur la bande est remplacé. Les attributs du nouvel ensemble de données sont définis à partir des attributs transmis dans la définition de transfert. Si aucun attribut n'est spécifié, les attributs sont définis sur les mêmes attributs que ceux de l'ensemble de données source ou sur les valeurs par défaut lorsque la source est un fichier. Les attributs d'un fichier bande existant sont ignorés.

Transfert à partir d'un fichier vers un ensemble de données-transferts binaires

Le format du fichier de destination détermine la longueur de l'enregistrement de destination. Vérifiez que le fichier existe sur le système de destination ou spécifiez le fichier de destination avec les attributs appropriés afin que le fichier soit créé correctement. Si vous ne spécifiez pas d'attributs, le système spécifie la valeur par défaut suivante: un fichier séquentiel physique avec un format d'enregistrement non défini et la taille de bloc maximale (BLKSIZE) de l'unité (renvoyée par la macro DEVTYPE). Par exemple, pour une unité de stockage à accès direct (DASD), la taille est 6144 et pour une bande, la taille est 32760. Si vous souhaitez transférer un fichier sur un système réparti vers un ensemble de données z/OS en mode binaire, notez le comportement suivant:

Fichiers de destination physique séquentiels (PS):

- Le fichier source sur le système réparti est lu séquentiellement pour remplir chaque enregistrement ou bloc.
- Dans les jeux de données de format variable, chaque enregistrement est rempli à sa capacité.

Fichiers de destination de fichier partitionné (PDS):

- Chaque fichier source est copié dans un membre PDS portant le même nom ou un nom équivalent. Si le nom de fichier dépasse la longueur maximale autorisée pour un nom de membre, le nom de fichier est converti en nom de membre valide. Pour plus d'informations sur les noms de membre, voir [Conventions de dénomination d'objet](#). Si le fichier source est un répertoire, chaque fichier de ce répertoire devient membre de l'ensemble de données partitionnées.
- Si un membre PDS existe, le membre est remplacé si vous avez spécifié le remplacement des fichiers de destination existants pour le transfert. Si vous ne spécifiez pas d'écrasement, le transfert échoue.
- Le fichier source du système réparti est lu séquentiellement pour remplir chaque enregistrement ou bloc du membre.
- Sur les membres PDS de format variable, chaque enregistrement est rempli à sa capacité.

Transfert d'un fichier vers un ensemble de données-transferts de texte

Le format du fichier de destination détermine la longueur de l'enregistrement de destination. Vérifiez que le fichier existe sur le système de destination ou spécifiez le fichier de destination avec les attributs appropriés afin que le fichier soit créé correctement. Si vous souhaitez effectuer un transfert depuis un fichier d'un système réparti vers un fichier z/OS sous forme de texte, notez le comportement suivant:

Fichiers de destination physique séquentiels (PS):

- Chaque ligne de texte devient un enregistrement (ou un bloc pour les fichiers de format d'enregistrement non défini (RECFM = U)). Les caractères de fin de ligne ne sont pas présents dans les enregistrements de fichier (pour les fichiers non ASA uniquement).
- Lorsque des caractères de contrôle de format ASA sont utilisés dans le fichier de destination, les caractères de fin de ligne sont effectivement convertis en code de contrôle de format ASA équivalent.
- Lorsqu'une ligne est plus longue qu'un enregistrement, elle est divisée à la limite de l'enregistrement et passe à l'enregistrement suivant.

Fichiers de destination PDS:

- Chaque fichier source est copié dans un membre PDS portant le même nom ou un nom équivalent. Si le nom de fichier dépasse la longueur maximale autorisée pour un nom de membre, le nom de fichier est converti en nom de membre valide. Pour plus d'informations sur les noms de membre, voir [Conventions de dénomination d'objet](#). Si le fichier source est un répertoire, chaque fichier de ce répertoire devient membre de l'ensemble de données partitionnées.
- Si un membre PDS existe, le membre est remplacé si vous avez spécifié le remplacement des fichiers de destination existants pour le transfert. Si vous ne spécifiez pas d'écrasement, le transfert échoue.
- Chaque ligne de texte devient un enregistrement (ou un bloc pour les fichiers de format d'enregistrement non défini (RECFM = U)). Les caractères de fin de ligne ne sont pas présents dans les enregistrements de membre (pour les fichiers non ASA uniquement).
- Lorsque des caractères de contrôle de format ASA sont utilisés dans le fichier de destination, les caractères de fin de ligne sont effectivement convertis en code de contrôle de format ASA équivalent.
- Lorsqu'une ligne est plus longue qu'un enregistrement, elle est divisée à la limite de l'enregistrement et passe à l'enregistrement suivant.

Transfert à partir d'un ensemble de données vers un fichier-transferts binaires et texte

Si vous souhaitez effectuer un transfert d'un ensemble de données vers un fichier en tant que fichier binaire ou texte, notez le comportement suivant:

- Le contenu de chaque enregistrement est transféré sous forme binaire dans un fichier ; aucun enregistrement, aucune information de format de bloc ou aucun caractère de contrôle de format ASA n'est transféré.
- Pour les transferts de texte uniquement, chaque enregistrement de fichier devient une ligne dont le texte est converti dans la page de codes de l'agent de destination. Autrement dit, un retour chariot-saut de ligne (CRLF) est ajouté pour un système de destination Windows et un retour chariot (CR) est ajouté pour un système de destination UNIX .
- **Jeux de données source non VSAM et PS.** Les enregistrements du fichier source sont transférés vers le fichier cible et concaténés. Si le fichier de destination existe, il est remplacé, en fonction de l'option de comportement du fichier de destination que vous avez spécifiée pour le transfert de fichier. Si la destination est spécifiée en tant que répertoire plutôt qu'en tant que fichier, le nom du fichier de destination sera le nom de l'ensemble de données, à l'exclusion du qualificatif de haut niveau (HLQ).
- **Jeux de données source PDS.** Chaque membre spécifié, ou tous les membres si aucun membre n'est spécifié, est extrait vers la destination. Si la destination indique un répertoire, les membres sont extraits dans des fichiers distincts. Sinon, chaque membre spécifié est écrit dans le fichier de destination, ce qui entraîne le transfert d'un seul membre. Si le fichier de destination existe pour un membre, le fichier est remplacé, en fonction de l'option de comportement du fichier de destination que vous avez spécifiée pour le transfert de fichier.

Référence associée

«Instructions pour le transfert de fichiers», à la page 648

Selon le système d'exploitation depuis et vers lequel vous effectuez le transfert et selon que vous effectuez le transfert en mode binaire ou texte, il existe des instructions sur le comportement à attendre.

«Transfert entre fichiers sous z/OS», à la page 651

Vous pouvez effectuer un transfert entre des fichiers z/OS à l'aide de Managed File Transfer. Examinez attentivement le comportement suivant pour vous assurer que vos fichiers sont transférés correctement.

«fteCreateTransfer: démarrer un nouveau transfert de fichier», à la page 447

La commande **fteCreateTransfer** crée et démarre un nouveau transfert de fichier à partir de la ligne de commande. Cette commande peut démarrer un transfert de fichier immédiatement, planifier un transfert de fichier à une heure et une date ultérieures, répéter un transfert planifié une ou plusieurs fois et déclencher un transfert de fichiers sous certaines conditions.

Transfert entre fichiers sous z/OS

Vous pouvez effectuer un transfert entre des fichiers z/OS à l'aide de Managed File Transfer. Examinez attentivement le comportement suivant pour vous assurer que vos fichiers sont transférés correctement.

Managed File Transfer ne prend pas en charge les fichiers décatalogués sur disque ou sur bande. Les fichiers existants doivent être catalogués et les nouveaux fichiers doivent être catalogués.

Prenez en compte les cas suivants:

Si vous copiez ou déplacez un fichier entre des systèmes z/OS et que la destination n'existe pas.

Par défaut, le fichier de destination est créé avec les mêmes caractéristiques que la source. Vous pouvez spécifier des attributs pour le fichier de destination afin de remplacer les caractéristiques par défaut. Dans ce cas, une vérification de compatibilité est effectuée pour s'assurer que le transfert est possible.

Si vous copiez ou déplacez un fichier entre des systèmes z/OS et que la destination existe déjà.

- Si vous spécifiez des attributs pour le fichier de destination afin de remplacer les caractéristiques par défaut, une vérification de compatibilité est effectuée pour s'assurer que le fichier de destination est accessible de la manière requise. Toutefois, vous ne pouvez pas remplacer les attributs suivants:
 - Organisation et type de l'ensemble de données de base
 - Longueur d'enregistrement logique (LRECL)
 - Taille de bloc (BLKSIZE)

Si vous transférez un fichier sur bande.

Lorsque vous transférez un fichier sur bande, tout fichier existant déjà sur la bande est remplacé. Les attributs du nouvel ensemble de données sont définis à partir des attributs transmis dans la définition de transfert. Si aucun attribut n'est spécifié, les attributs sont définis sur les mêmes valeurs que celles de l'ensemble de données source ou sur les valeurs par défaut lorsque la source est un fichier. Les attributs d'un fichier bande existant sont ignorés.

En outre, l'ID utilisateur sous lequel l'agent de destination s'exécute doit disposer des droits appropriés pour monter les bandes. Pour plus d'informations sur la procédure à suivre, reportez-vous à la documentation du gestionnaire de sécurité externe utilisé par votre entreprise.

Si vous effectuez un transfert depuis une bande vers un fichier.

Pour accéder à un fichier sur bande, l'ID utilisateur que l'agent source exécute doit disposer des droits appropriés pour monter les bandes. Pour plus d'informations sur la procédure à suivre, reportez-vous à la documentation du gestionnaire de sécurité externe utilisé par votre entreprise.

Compatibilité des fichiers

Passez en revue le comportement et les restrictions suivants pour la compatibilité des fichiers:

Différences de format et de longueur d'enregistrement:

- Les enregistrements à format variable utilisent une zone de longueur d'enregistrement de 4 octets dans les données d'enregistrement. Par conséquent, pour un transfert d'un enregistrement fixe vers un fichier d'enregistrement variable, la longueur d'enregistrement variable doit être supérieure ou égale à la longueur d'enregistrement fixe plus 4. Pour un transfert d'un fichier d'enregistrement à format variable vers un fichier d'enregistrement à format fixe, la longueur d'enregistrement du fichier d'enregistrement à format fixe doit être supérieure ou égale à la longueur d'enregistrement variable moins 4.

Différences de taille de bloc:

- Pour les données d'enregistrement à format fixe et à format variable, les différences de taille de bloc rendent la présentation du jeu de données source et de destination différente.
- Pour les enregistrements de format non définis, à condition que la taille de bloc de destination soit supérieure ou égale à la taille de bloc du fichier source, vous pouvez transférer un fichier.
- Pour les fichiers de format non définis, vous ne pouvez pas effectuer de transfert si la taille de bloc source est supérieure à la taille de bloc de destination.

Fichiers partitionnés (PDS) et fichiers partitionnés étendus (PDSE)

Le comportement et les restrictions suivants s'appliquent également aux ensembles de données partitionnées et aux ensembles de données partitionnées étendu:

- Si vous transférez un membre PDS ou PDSE vers un PDS ou PDSE de destination, un membre du PDS ou PDSE de destination est créé. Si le membre PDS ou PDSE de destination existe déjà, le membre est remplacé. Si vous transférez un membre PDS ou PDSE vers un fichier de destination non PDS ou non PDSE, le fichier de destination est créé pour contenir les données du membre. Si le fichier de destination existe déjà, il est remplacé.
- Si vous tentez de transférer un ensemble de données partitionnées (PDS) ou un ensemble de données partitionnées (PDSE) vers une destination non-PDS ou non-PDSE, tous les membres de l'ensemble de données partitionnées (PDS) ou de l'ensemble de données partitionnées (PDSE) sont écrits dans la destination non-PDSE. Chaque transfert de membre suivant écrase le contenu précédent de la destination non PDSE ou échoue, en fonction des options de transfert.
- Lorsque vous transférez un PDS ou un PDSE vers un PDS ou un PDSE de destination, une copie de l'ensemble du PDS ou du PDSE est créée au niveau de la destination. Si le PDS ou le PDSE de destination existe déjà, les membres de la source sont ajoutés. Si un membre PDS ou PDSE existe déjà sur la destination, le membre est remplacé.
- Le transfert d'un ensemble de données partitionnées ou non partitionnées ou non partitionnées vers un ensemble de données partitionnées ou partitionnées cible ajoute le contenu de l'ensemble de données partitionnées ou non partitionnées en tant que nouveau membre de l'ensemble de données partitionnées ou partitionnées partitionnées. Si le membre PDS existe déjà, il est remplacé. Si vous ne spécifiez pas de nom pour un nouveau membre, un nom est généré à partir du fichier source ou du nom symbolique.
- Il existe une limitation connue avec les transferts vers les fichiers PDS et PDSE sur les systèmes où l'espace disque est limité. Pour plus de détails, voir la section [z/OS dans Problèmes communs MFT](#).
- **Remarque :** Lorsque vous transférez un PDS ou un PDSE vers un PDS ou un PDSE cible, les informations et les statistiques du membre ne sont pas conservées. Par exemple, si vous transférez une bibliothèque de chargement stockée en tant que PDS, le PDS de destination n'est pas utilisable en tant que bibliothèque de chargement.

Transferts binaires et texte

Le transfert binaire pour les fichiers est défini comme étant les données d'enregistrement dans leur forme binaire, telles qu'elles sont lues à partir du fichier à l'aide du format d'enregistrement par défaut (type = enregistrement). Les données sont lues et écrites enregistrement par enregistrement. Le service système effectue la conversion d'enregistrement et de bloc nécessaire (où les ensembles de données ont des

paramètres d'enregistrement et de bloc différents) et la conversion d'ASA et de code de contrôle machine nécessaire. Si un fichier est défini pour les caractères de contrôle de format ASA et que l'autre n'est pas approprié, la conversion en codes de contrôle normaux est effectuée à l'aide du comportement de la fonction de bibliothèque système C/C + +.

Groupes de données de génération (GDG)

Managed File Transfer prend en charge les groupes de données de génération (GDG) pour les fichiers source et de destination sur z/OS. Les noms d'ensemble de fichiers absolus et relatifs sont pris en charge. Lorsque vous écrivez dans une nouvelle génération, l'ensemble de fichiers de base doit déjà exister.

Référence associée

«Instructions pour le transfert de fichiers», à la page 648

Selon le système d'exploitation depuis et vers lequel vous effectuez le transfert et selon que vous effectuez le transfert en mode binaire ou texte, il existe des instructions sur le comportement à attendre.

«Transfert de groupes de données de génération (GDG)», à la page 672

Managed File Transfer prend en charge les groupes de données de génération (GDG) pour les fichiers source et de destination sur z/OS. Les noms d'ensemble de fichiers absolus et relatifs sont pris en charge. Lorsque vous écrivez dans une nouvelle génération, l'ensemble de fichiers de base doit exister.

«Transfert de fichiers vers et depuis des noeuds Connect:Direct», à la page 653

Vous pouvez transférer des ensembles de données entre les agents Managed File Transfer et les noeuds IBM Sterling Connect:Direct à l'aide du pont Connect:Direct . Vous pouvez spécifier un fichier comme source de transfert, destination de transfert ou les deux.

«Transfert de fichiers et de jeux de données entre z/OS et les systèmes répartis», à la page 649

Vous pouvez transférer des fichiers et des types d'ensemble de données pris en charge entre z/OS et les systèmes de fichiers distribués à l'aide de Managed File Transfer. Examinez attentivement le comportement suivant, qui dépend du type de système depuis et vers lequel vous effectuez le transfert.

Transfert de fichiers vers et depuis des noeuds Connect:Direct

Vous pouvez transférer des ensembles de données entre les agents Managed File Transfer et les noeuds IBM Sterling Connect:Direct à l'aide du pont Connect:Direct . Vous pouvez spécifier un fichier comme source de transfert, destination de transfert ou les deux.

Spécification des noms de fichier

Pour spécifier un jeu de données sur un noeud Connect:Direct dans une demande de transfert, utilisez la syntaxe utilisée pour les transferts de jeu de données entre les agents Managed File Transfer , mais avec deux modifications:

- Vous devez faire précéder le nom de l'ensemble de données du nom de noeud Connect:Direct et d'un signe deux-points (:). La syntaxe est la suivante:

```
cdNode:data_set_name{;attrib1;...;attribN}
```

Par exemple, pour spécifier un fichier partitionné appelé OBJECT.LIB sur le système où se trouve le noeud Connect:Direct CD_NODE1 , utilisez la syntaxe suivante:

```
CD_NODE1: //'OBJECT.LIB';RECFM(F,B);BLKSIZE(800);LRECL(80)
```

Dans cet exemple, trois attributs facultatifs sont spécifiés par le texte RECFM(F,B);BLKSIZE(800);LRECL(80).

- Le nom de fichier spécifié est interprété comme un nom de fichier qualifié complet, qu'il soit placé entre apostrophes ou non. Le système n'ajoute jamais de préfixe. Si vous souhaitez spécifier un préfixe, tel que l'ID utilisateur sous lequel l'agent s'exécute, vous devez le spécifier dans le nom de l'ensemble de données. Cela diffère du comportement des transferts de fichier qui impliquent uniquement des agents

Managed File Transfer , où si le nom de fichier spécifié n'est pas encadré par des guillemets simples, le système ajoute un préfixe du qualificatif de haut niveau par défaut pour l'agent cible.

A l'exception de ces deux modifications, spécifiez le nom de fichier et les attributs facultatifs à l'aide de la même syntaxe que celle utilisée pour les transferts de fichier entre les agents Managed File Transfer , qui comporte les règles suivantes:

- Vous devez préfixer le nom de l'ensemble de données avec deux barres obliques (//).
- Si vous souhaitez spécifier des attributs de fichier, indiquez-les après le nom de fichier, en les séparant par des points-virgules. Les attributs doivent être fournis au format *key (value)* , qui convient à BPXWDYN.

Pour plus d'informations sur la spécification de fichiers dans une demande de transfert, voir [«fteCreateTransfer: démarrer un nouveau transfert de fichier»](#), à la page 447 et [«fteCreateTemplate: créer un nouveau modèle de transfert de fichiers»](#), à la page 431.

Paramètres à utiliser dans votre demande de transfert

Pour la plupart des demandes de transfert impliquant des ensembles de données sur les noeuds Connect:Direct , vous pouvez spécifier les ensembles de données source et de destination de la même manière que pour un transfert d'ensemble de données impliquant uniquement des agents Managed File Transfer . Utilisez les paramètres **source_specification**, **-dset -dp** avec les commandes **fteCreateTransfer** ou **fteCreateTemplate** . Cette syntaxe est prise en charge pour les scénarios suivants:

- Tous les agents impliqués dans le transfert sont IBM WebSphere MQ File Transfer Edition 7.0.4 ou version ultérieure
- L'agent source est l'agent de pont Connect:Direct et est donc IBM WebSphere MQ File Transfer Edition 7.0.4 ou version ultérieure, et l'agent de destination est IBM WebSphere MQ File Transfer Edition 7.0.3 ou version antérieure.

Si l'agent de destination est l'agent de pont Connect:Direct et que l'agent source est IBM WebSphere MQ File Transfer Edition 7.0.3 ou une version antérieure, vous devez apporter les modifications suivantes à votre demande de transfert:

- Pour spécifier un membre de fichier séquentiel ou de fichier partitionné (PDS) comme destination d'un transfert, utilisez le paramètre **-df** .
- Pour spécifier un fichier partitionné comme destination d'un transfert, utilisez le paramètre **-dd** .

Vous pouvez également utiliser cette syntaxe comme alternative aux paramètres **-ds** et **-dp** habituels pour les transferts où l'agent source est IBM WebSphere MQ File Transfer Edition 7.0.4 ou version ultérieure. Par exemple, si vous souhaitez utiliser une syntaxe cohérente dans tous vos scénarios et que certains scénarios impliquent un agent source IBM WebSphere MQ File Transfer Edition 7.0.3 ou antérieur, utilisez les paramètres **-df** et **-dd** .

Remarque : Si la destination du transfert est un PDS et que l'agent de destination est l'agent de pont Connect:Direct , vous devez spécifier le paramètre **-de** avec la valeur `overwrite`.

Spécification des attributs de fichier

Certains attributs de fichier sont définis par Managed File Transfer et transmis en tant que paramètres au processus Connect:Direct **COPY** . Vous pouvez également fournir certains attributs dans la demande de transfert en spécifiant la clé BPXWDYN appropriée. Le pont Connect:Direct convertit les clés ayant des propriétés Connect:Direct équivalentes au format requis par Connect:Direct. Par exemple, dans la spécification de fichier `CD_NODE1: // ' OBJECT . LIB ' ; RECFM(F, B) ; BLKSIZE(800) ; LRECL(80)`, les attributs `RECFM(F, B) ; BLKSIZE(800) ; LRECL(80)` sont convertis en `DCB=(RECFM=FB, BLKSIZE=800, LRECL=80)`.

Pour plus de détails sur les mappages entre ces deux types de paramètre, y compris les détails des clés BPXWDYN prises en charge pour une utilisation avec un transfert Connect:Direct , voir [«Mappages entre les paramètres d'instruction de processus Connect:Direct et les clés BPXWDYN»](#), à la page 655. Toutes

les clés BPXWDYN n'ont pas de paramètre de processus Connect:Direct équivalent et tous les paramètres de processus Connect:Direct n'ont pas de clé BPXWDYN équivalente.

Autres considérations

- Si votre destination de transfert est un fichier partitionné sur un noeud Connect:Direct , vous devez créer le fichier partitionné avant le transfert, car le noeud Connect:Direct ne le crée pas pour vous.

Concepts associés

«Le pont Connect:Direct», à la page 236

Vous pouvez transférer des fichiers depuis et vers un réseau IBM Sterling Connect:Direct existant. Utilisez le pont Connect:Direct, qui est un composant de Managed File Transfer, pour transférer des fichiers entre MFT et IBM Sterling Connect:Direct.

Tâches associées

 Transfert d'un fichier vers un noeud Connect:Direct sous z/OS

Vous pouvez transférer un ensemble de données d'un agent Managed File Transfer sous z/OS vers un noeud Connect:Direct sous z/OS à l'aide d'un pont Connect:Direct situé sur un système Windows ou Linux .

Référence associée

«Transfert entre fichiers sous z/OS», à la page 651

Vous pouvez effectuer un transfert entre des fichiers z/OS à l'aide de Managed File Transfer. Examinez attentivement le comportement suivant pour vous assurer que vos fichiers sont transférés correctement.

«fteCreateTransfer: démarrer un nouveau transfert de fichier», à la page 447

La commande **fteCreateTransfer** crée et démarre un nouveau transfert de fichier à partir de la ligne de commande. Cette commande peut démarrer un transfert de fichier immédiatement, planifier un transfert de fichier à une heure et une date ultérieures, répéter un transfert planifié une ou plusieurs fois et déclencher un transfert de fichiers sous certaines conditions.

«fteCreateTemplate: créer un nouveau modèle de transfert de fichiers», à la page 431

La commande **fteCreateTemplate** crée un modèle de transfert de fichier que vous pouvez conserver pour une utilisation ultérieure. Le seul paramètre obligatoire est le paramètre **-tn** (*template_name*). Tous les autres paramètres sont facultatifs, bien que si vous spécifiez une spécification de fichier source, vous devez également fournir un fichier de destination. De même, si vous spécifiez un fichier de destination, vous devez également spécifier une spécification de fichier source.

«Chemins d'accès aux fichiers Connect:Direct spécifiés avec une double barre oblique», à la page 357

Si, dans le cadre d'un transfert de fichier, vous spécifiez un fichier situé sur un noeud Connect:Direct en utilisant un chemin de fichier commençant par une double barre oblique (/), le fichier est traité comme un jeu de données.

Mappages entre les paramètres d'instruction de processus Connect:Direct et les clés BPXWDYN

Lorsque vous soumettez une demande de transfert pour un fichier dont la source ou la destination est un noeud Connect:Direct , toutes les clés BPXWDYN prises en charge que vous fournissez sont converties dans un format accepté par les processus Connect:Direct .

Pour plus d'informations sur les instructions de processus IBM Sterling Connect:Direct , voir le manuel Connect:Direct [Process Language Reference Guide](#) .

Tableau 58. Paramètres de l'instruction Connect:Direct **COPY** et clés BPXWDYN équivalentes utilisées par Managed File Transfer

Paramètre de l'instruction Connect:Direct COPY	Clé BPXWDYN
DSN	DSN (valide pour les transferts vers et depuis des fichiers). La spécification de cette clé remplace la valeur de paramètre affectée par Managed File Transfer, qui est basée sur les spécifications de fichier source ou de destination fournies dans la demande de transfert.
Fichier	Pas de mappage pour les fichiers.
PNODE	Pas de mappage: Le noeud principal du transfert est identifié par Managed File Transfer. Si vous tentez de fournir une valeur pour ce paramètre, une erreur est générée.
SNODE	Pas de mappage: Le noeud secondaire pour le transfert est identifié par Managed File Transfer. Si vous tentez de fournir une valeur pour ce paramètre, une erreur est générée.
Bloc de contrôle des données	Voir Mappages pour les sous-paramètres de DCB
DECISION	Voir Mappages pour les sous-paramètres de DISP pour une instruction COPY From et Mappages pour les sous-paramètres de DISP pour une instruction COPY To
RESGDG	Pas de mappage
Libellé	Voir Mappages pour les sous-paramètres de LABEL
MSVGP	Pas de mappage
UNIT	UNIT
VOL	Voir Mappages pour les sous-paramètres de VOL
Alias	Pas de mappage
EXCLUDE	Pas de mappage
PDS.DIR	Pas de mappage. Managed File Transfer définit la valeur de ce paramètre de processus sur N, de sorte qu'aucune information liée à l'utilisateur qui se trouve dans le répertoire n'est envoyée.
REPLACEZ NOREPLACE	Aucun équivalent BPXWDYN. Le comportement lorsqu'un fichier de destination existe déjà sur le système de destination est défini par la valeur du paramètre -de (destination_file_behavior) dans la demande de transfert. Pour plus d'informations sur le comportement par défaut de Managed File Transfer lorsqu'un fichier de destination existe déjà, voir « Transfert entre fichiers sous z/OS », à la page 651.
SELECT	Aucun équivalent BPXWDYN. Les membres de fichier sélectionnés pour la copie sont définis par la spécification de fichier source dans la demande de transfert.

Tableau 58. Paramètres de l'instruction Connect:Direct **COPY** et clés BPXWDYN équivalentes utilisées par Managed File Transfer (suite)

Paramètre de l'instruction Connect:Direct COPY	Clé BPXWDYN
BUFND	Pas de mappage
IOEXIT	Pas de mappage
EXIT	Pas de mappage
SYSOPTS	Voir Mappages pour les sous-paramètres de SYSOPTS
TYPE	Pas de mappage
AVGREC	Pas de mappage
Dataclas	Dataclas
TYPE DE NOM	DSNTYPE. La spécification d'une valeur de PDS pour cette clé remplace la valeur de paramètre affectée par Managed File Transfer, qui est LIBRARY. Il n'existe aucun mappage pour une autre valeur-EXTPREF, EXTREQ, BASIC ou LARGE. La spécification de l'une de ces valeurs non prises en charge génère une erreur. La spécification de PDS ou LIBRARY pour un fichier séquentiel génère une erreur.
KEYLEN	Pas de mappage
DESACTIVEE	Pas de mappage
SIMILAIRE A	SIMILAIRE A
LRECL	Pas de mappage
Mgmtclas	Mgmtclas
RECORG	Pas de mappage
SECMODELE	Pas de mappage
Storclas	Storclas
Espace	Voir Mappages pour les sous-paramètres de SPACE
SYSOUT	Pas de mappage
CKPT	Pas de mappage
COMPRESS	Pas de mappage
Sécurisé	Pas de mappage

Tableau 59. Sous-paramètres du paramètre **DCB** pour l'instruction Connect:Direct **COPY** et clés BPXWDYN équivalentes utilisées par Managed File Transfer

Sous-paramètres du paramètre DCB	Clé BPXWDYN
nom-fichier-modèle	Pas de mappage
Taille de bloc	Taille de bloc
Network Control Program	BUFNO
DEN	Pas de mappage
DSORG	DSORG

Tableau 59. Sous-paramètres du paramètre **DCB** pour l'instruction Connect:Direct **COPY** et clés BPXWDYN équivalentes utilisées par Managed File Transfer (suite)

Sous-paramètres du paramètre DCB	Clé BPXWDYN
KEYLEN	Pas de mappage
LIMCT	Pas de mappage
LRECL	LRECL
OPTCD	Pas de mappage
RECFM	RECFM
RKP	Pas de mappage
TRTCH	TRTCH

Tableau 60. Sous-paramètres du paramètre **DISP** pour l'instruction Connect:Direct **COPY From** et clés BPXWDYN équivalentes utilisées par Managed File Transfer

Sous-paramètres du paramètre DISP pour une instruction COPY From	Clé BPXWDYN	Détails
[OLD SHR]	[OLD SHR]	Indique le statut du fichier avant le transfert. Managed File Transfer définit ce sous-paramètre sur SHR .
[KEEP SUPPRIMER]	[KEEP DELETE] ou PATHDISP	Indique le statut du fichier une fois que le transfert a abouti. La valeur définie par Managed File Transfer dépend de la disposition du fichier source, définie par le paramètre -sd .
[KEEP SUPPRIMER]	[KEEP DELETE] ou PATHDISP	Indique le statut du fichier une fois le transfert terminé de manière anormale. Managed File Transfer définit ce sous-paramètre sur KEEP .

Tableau 61. Sous-paramètres du paramètre **DISP** pour l'instruction Connect:Direct **COPY To** et clés BPXWDYN équivalentes utilisées par Managed File Transfer

Sous-paramètres du paramètre DISP pour une instruction COPY To	Clé BPXWDYN	Détails
[NOUVEAU OLD MOD RPL SHR]	[NOUVEAU OLD MOD SHR]	Indique le statut du fichier avant le transfert. La valeur définie par Managed File Transfer dépend de la valeur du paramètre -de (destination_file_behavior) dans la demande de transfert. Si le fichier de destination n'existe pas déjà, la valeur du sous-paramètre est NEW . Si le fichier existe déjà, la valeur du sous-paramètre est RPL . Managed File Transfer ne prend pas en charge la clé RPL fournie dans une demande de transfert.
[KEEP CATLG]	[KEEP CATLOG] ou PATHDISP	Indique le statut du fichier une fois que le transfert a abouti. Managed File Transfer définit ce sous-paramètre sur CATLOG .
[KEEP CATLG DELETE]	[KEEP DELETE] ou PATHDISP	Indique le statut du fichier une fois le transfert terminé de manière anormale. Managed File Transfer définit ce sous-paramètre sur DELETE .

Tableau 62. Sous-paramètres du paramètre **LABEL** pour l'instruction Connect:Direct **COPY** et clés BPXWDYN équivalentes utilisées par Managed File Transfer

Sous-paramètres du paramètre LABEL pour une instruction COPY	Clé BPXWDYN	Détails
numéro-séquence-fichier	SEQUENCE	
[SL AL BLP LTM NL]	LABEL (<i>type</i>)	Les valeurs possibles de <i>type</i> sont NL, SL, NSL, SUL, BLP, LTM, AL et AUL. Connect:Direct accepte un sous-ensemble de ces valeurs. Si vous spécifiez une valeur qui n'est pas prise en charge par Connect:Direct, Connect:Direct génère un message d'erreur.
[MOT_DE_PASSE NOPWREAD]	Pas de mappage	
[IN OUT]	Pas de mappage	
[RETPD EXPDT]	RETPD	EXPDT non pris en charge

Tableau 63. Sous-paramètres du paramètre **VOL** pour l'instruction **Connect:Direct COPY** et clés BPXWDYN équivalentes utilisées par Managed File Transfer

Sous-paramètres du paramètre VOL pour une instruction COPY	Clé BPXWDYN
PRIVATE	Pas de mappage
CONSERVER	Pas de mappage
numéro-séquence-volume	Pas de mappage
volume-count	MAXVOL
Ser	VOL
REF	Pas de mappage

Tableau 64. Sous-paramètres du paramètre **SYSOPTS** pour l'instruction **Connect:Direct COPY** et clés BPXWDYN équivalentes utilisées par Managed File Transfer

Sous-paramètres du paramètre SYSOPTS pour une instruction COPY	Clé BPXWDYN
DBCS	Pas de mappage
PAGE DE CODES	La valeur dépend des options de transfert Managed File Transfer . Pour plus d'informations, voir «Transfert de fichiers texte avec MFT», à la page 662.
TYPE DE DONNEES	Pas de mappage. Managed File Transfer définit cette valeur sur TEXT pour les transferts de texte vers ou depuis un fichier, et sur BINARY dans le cas contraire.
XLATE	Pas de mappage. Managed File Transfer définit cette valeur sur NO lorsque la valeur de DATATYPE est TEXT.
STRIP.BLANKS	Pas de mappage. Managed File Transfer définit cette valeur sur YES lorsque la valeur de DATATYPE est TEXT.
PERMISS	Pas de mappage
PRECOMP	Pas de mappage
UNIQUE	Pas de mappage
SYSOUT	Pas de mappage

Tableau 65. Sous-paramètres du paramètre **SPACE** pour l'instruction **Connect:Direct COPY** et clés BPXWDYN équivalentes utilisées par Managed File Transfer

Sous-paramètres du paramètre SPACE pour une instruction COPY	Clé BPXWDYN
CYL	CYL
TRK	Pistes
BLK	Blocs
av-rec-len	Pas de mappage
prim, [sec], [dir]	SPACE (prim [, sec]), DIR

Tableau 65. Sous-paramètres du paramètre **SPACE** pour l'instruction **Connect:Direct COPY** et clés **BPXWDYN** équivalentes utilisées par Managed File Transfer (suite)

Sous-paramètres du paramètre SPACE pour une instruction COPY	Clé BPXWDYN
RLSE	LIBERER
CONTIG	Pas de mappage
ARRONDI	Pas de mappage

Concepts associés

«Le pont **Connect:Direct**», à la page 236

Vous pouvez transférer des fichiers depuis et vers un réseau IBM Sterling **Connect:Direct** existant. Utilisez le pont **Connect:Direct**, qui est un composant de Managed File Transfer, pour transférer des fichiers entre MFT et IBM Sterling **Connect:Direct**.

Tâches associées

 Transfert d'un fichier vers un noeud **Connect:Direct** sous z/OS

Vous pouvez transférer un ensemble de données d'un agent Managed File Transfer sous z/OS vers un noeud **Connect:Direct** sous z/OS à l'aide d'un pont **Connect:Direct** situé sur un système Windows ou Linux .

Référence associée

Transfert de fichiers depuis et vers des noeuds **Connect:Direct**

Vous pouvez transférer des ensembles de données entre les agents Managed File Transfer et les noeuds IBM Sterling **Connect:Direct** à l'aide du pont **Connect:Direct** . Vous pouvez spécifier un fichier comme source de transfert, destination de transfert ou les deux.

Propriétés **BPXWDYN** que vous ne devez pas utiliser avec MFT

Certaines options **BPXWDYN** ne doivent pas être spécifiées en cas d'utilisation de la commande **fteCreateTemplate**, de la commande **fteCreateTransfer** ou de la propriété **bpxwdynAllocAdditionalOptions** dans le fichier `agent.properties`.

Un certain nombre d'options **BPXWDYN** ne doivent pas être spécifiées avec Managed File Transfer car elles sont utilisées par l'agent ou ne sont pas prises en charge. Si vous utilisez ces options, elles peuvent entraîner un comportement imprévisible ; les options sont répertoriées dans le tableau suivant.

Options BPXWDYN	Description
DSN DA	Indique le nom de fichier à allouer.
DD FI	Indique le nom symbolique à allouer.
DONNEES DE LA CARTE	Indique aux services de méthode d'accès séquentiel si les données sont traitées comme du texte ou des données binaires.
ANCIEN MOD SHR NOUVEAU SYSOUT	Indique le statut de l'ensemble de données.
Réutiliser	Indique que le fichier nommé est libéré avant l'exécution de la fonction.
HOLD	Indique que le fichier de sortie doit être conservé jusqu'à sa libération par l'utilisateur ou l'opérateur.
CONSERVER SUPPRESSION CATALOGUE CATALOGUE NON CATALOGUE	Indique la disposition de l'ensemble de données après sa libération.
RECORG (LS)	Crée un fichier linéaire VSAM.
MSG	Dirige les messages d'allocation. Remarque : Cette option peut être utilisée, mais comme Managed File Transfer l'utilise pour diriger les informations d'erreur vers le journal de transfert, son utilisation peut entraîner un comportement imprévisible.

Transfert de fichiers texte avec MFT

Le transfert de fichiers texte implique la conversion de la page de codes d'un fichier d'une page de codes à une autre. Le transfert de fichier texte implique également la conversion de caractères CRLF (retour chariot-saut de ligne) entre les systèmes. Cette rubrique récapitule le comportement de transfert de fichier texte de Managed File Transfer.

Sauf indication contraire, la conversion provient de la page de codes par défaut du système source du fichier vers la page de codes par défaut de son système cible. En outre, le transfert de fichier texte effectue une conversion de nouvelle ligne, ce qui signifie que les caractères de nouvelle ligne du fichier de destination sont ceux qui sont natifs de sa plateforme de destination. Vous pouvez remplacer l'utilisation des pages de codes par défaut sur un système en spécifiant la page de codes à utiliser pour la lecture du fichier source et l'écriture du fichier de destination. Vous pouvez également spécifier la séquence de caractères de fin de ligne à utiliser pour le fichier de destination. Pour plus d'informations, voir les rubriques «[fteCreateTransfer: démarrer un nouveau transfert de fichier](#)», à la page 447 et «[Utilisation de fichiers de définition de transfert](#)», à la page 164.

Les transferts de fichiers texte effectuent des substitutions de point de code simples entre les pages de codes. Les transferts de fichiers texte n'effectuent pas de transferts ou de traductions complexes de données, par exemple, des conversions entre des formes visuelles et logiques de données bidirectionnelles ou de mise en forme de texte.

Zone	Comportement par défaut	Pouvez-vous changer ce comportement?
Codage du fichier source	Codage de la plateforme source	Oui Lorsque vous spécifiez un codage de fichier source et que la source est un jeu de données, le codage doit être une page de codes EBCDIC, sinon le transfert échoue. De même, si la destination est un fichier, le codage de la destination doit être une page de codes EBCDIC.
Séquence de caractères de fin de ligne du fichier source	Conversion d'une séquence unique (LF) ou CRLF (CRLF) en séquence de caractères de fin de ligne de destination	Non

Tableau 66. Comportement de transfert de fichier texte pour toutes les plateformes (suite)

Zone	Comportement par défaut	Pouvez-vous changer ce comportement?
Codage du fichier cible	Codage de la plateforme de destination	Oui Lorsque vous spécifiez un codage de fichier source et que la source est un jeu de données, le codage doit être une page de codes EBCDIC, sinon le transfert échoue. De même, si la destination est un fichier, le codage de la destination doit être une page de codes EBCDIC.
Séquence de caractères de fin de ligne du fichier cible	Fin de vie de la plateforme de destination	Oui
Séquence de caractères de remplacement de texte pour les caractères non mappables ou incorrectement formés dans la source ou la destination	Blanc, ce qui signifie que le transfert échoue si des caractères non mappables ou des caractères incorrectement formés sont présents. Vous pouvez utiliser la propriété <code>textReplacementCharacterSequence</code> pour spécifier le texte de remplacement, qui est décrit dans «Le fichier MFT <code>agent.properties</code> », à la page 564.	Oui

z/OS jeux de données



Lorsque des enregistrements de fichier sont accessibles en mode texte, chaque enregistrement représente une seule ligne. Les caractères de retour à la ligne n'existent pas dans l'enregistrement, mais pour les fichiers de format ASA, un caractère de code de contrôle de format ASA est défini pour représenter une nouvelle ligne (ou un autre caractère de contrôle). Lorsqu'une ligne de texte comportant un caractère de fin de nouvelle ligne est écrite dans un enregistrement, le caractère de nouvelle ligne est supprimé automatiquement ou un code de contrôle ASA approprié est défini, selon le cas. Lorsqu'un enregistrement est lu, un caractère de nouvelle ligne est automatiquement ajouté aux données renvoyées. Pour les jeux de données au format ASA, ce caractère peut être constitué de plusieurs nouvelles lignes ou d'un saut de page, selon le code de contrôle ASA de l'enregistrement.

En outre, pour les jeux de données à format fixe lorsqu'un enregistrement est lu, la nouvelle ligne est ajoutée après le dernier caractère de l'enregistrement qui n'est pas un caractère espace, ce qui rend les jeux de données à format fixe adaptés au stockage de texte.

Tableau 67. Comportement de transfert de fichier texte supplémentaire spécifique à z/OS

Zone	Comportement par défaut	Pouvez-vous changer ce comportement?
Longueur de ligne maximale	Paramètre <code>LRECL</code> ou <code>BLKSIZE</code> du fichier de destination, selon le cas	Non

Tableau 67. Comportement de transfert de fichier texte supplémentaire spécifique à z/OS (suite)

Zone	Comportement par défaut	Pouvez-vous changer ce comportement?
Retour à la ligne sur les lignes de longueur	Retour à la ligne. La ligne est divisée en plusieurs enregistrements et blocs selon les besoins.	Non

Lorsque l'agent Managed File Transfer est exécuté, la variable d'environnement `_EDC_ZERO_RECLEEN` est toujours définie sur "Y". Ce paramètre fait en sorte que le comportement de transfert de texte Managed File Transfer est identique à celui de FTP pour les jeux de données à blocs fixes et variables. Toutefois, pour les fichiers de format non définis, Managed File Transfer convertit les lignes d'espace unique en ligne vide et conserve les lignes vides. FTP convertit les lignes vides en lignes à espace unique et conserve les lignes à espace unique. Le tableau 3 décrit le comportement de Managed File Transfer et les différences de comportement FTP.

Le format du jeu de données détermine également la manière dont chaque ligne de texte est écrite dans un enregistrement. Pour les fichiers au format non ASA, les caractères de retour à la ligne et de retour chariot ne sont pas écrits dans l'enregistrement. Pour les jeux de données au format ASA, le premier octet de chaque enregistrement est un code de contrôle ASA représentant la fin des lignes, un saut de page et d'autres codes, selon le cas. Etant donné que les codes de contrôle ASA sont au début de chaque enregistrement, si le fichier texte source ne commence pas par une nouvelle séquence de caractères de ligne, une séquence de caractères de contrôle ASA vide (") (qui correspond à une nouvelle ligne) est insérée. Cela signifie que si le fichier ASA est transféré dans un fichier, une ligne vide est présente au début du fichier.

Tableau 68. Comportement de Managed File Transfer pour les fichiers

Format de fichier	Ligne de texte d'origine dans le fichier	Enregistrement de fichier	Lecture de l'enregistrement de fichier	Comportement de lecture FTP
Bloc fixe	Ligne vide	Enregistrement rempli d'espace	Ligne vide	Identique à MFT
Bloc fixe	Espace unique	Enregistrement rempli d'espace	Ligne vide	Identique à MFT
Bloc variable	Ligne vide	Enregistrement vide	Ligne vide	Identique à MFT
Bloc variable	Espace unique	Enregistrement d'espace unique	Espace unique	Identique à MFT
Non défini	Ligne vide	Enregistrement d'espace unique	Ligne vide	Espace unique
Non défini	Espace unique	Enregistrement d'espace unique	Ligne vide	Espace unique

Référence associée

«Instructions pour le transfert de fichiers», à la page 648

Selon le système d'exploitation depuis et vers lequel vous effectuez le transfert et selon que vous effectuez le transfert en mode binaire ou texte, il existe des instructions sur le comportement à attendre.

«Transfert de fichiers texte entre Connect:Direct et MFT», à la page 665

Le transfert de texte implique la conversion de la page de codes d'un fichier d'une page de codes à une autre. Le transfert de fichiers texte implique également la conversion de caractères CRLF (saut de ligne de retour chariot) entre les systèmes. Cette rubrique récapitule le comportement du transfert de fichiers texte dans les transferts entre un Managed File Transfer Agent et un noeud Connect:Direct .

«Pages de codes disponibles pour MFT», à la page 705

Cette rubrique de référence répertorie tous les formats de codage de caractères disponibles pour la conversion de fichier texte sur les différentes plateformes prises en charge par Managed File Transfer.

Transfert de fichiers texte entre Connect:Direct et MFT

Le transfert de texte implique la conversion de la page de codes d'un fichier d'une page de codes à une autre. Le transfert de fichiers texte implique également la conversion de caractères CRLF (saut de ligne de retour chariot) entre les systèmes. Cette rubrique récapitule le comportement du transfert de fichiers texte dans les transferts entre un Managed File Transfer Agent et un noeud Connect:Direct .

Pour plus d'informations sur le comportement des transferts de texte dans Managed File Transfer, voir «Transfert de fichiers texte avec MFT», à la page 662.

- Vérifiez que la mappe de réseau du noeud de pont Connect:Direct et tous les noeuds Connect:Direct utilisés comme destination de transfert incluent la description de plateforme appropriée.
 - Si votre noeud de pont Connect:Direct se trouve sur un système Windows, assurez-vous de sélectionner la valeur correcte dans la liste **Système d'exploitation** pour chaque noeud distant.
 - Si le noeud distant se trouve sur un système Windows, sélectionnez Windows.
 - Si le noeud distant se trouve sur un système UNIX ou Linux, sélectionnez UNIX.
 -  Si le noeud distant se trouve sur un système z/OS, sélectionnez OS/390.

Les transferts sur des noeuds distants sur d'autres systèmes d'exploitation ne sont pas pris en charge par le pont Connect:Direct.

- Vérifiez que, pour chaque nœud à distance vers ou depuis lequel vous transférez un fichier, indiquez le type de système d'exploitation du nœud Connect:Direct à distance dans le fichier `ConnectDirectNodeProperties.xml` du répertoire de configuration de l'agent de pont Connect:Direct. Pour plus d'informations, voir «Configurez le fichier `ConnectDirectNodeProperties.xml` pour inclure des informations sur les noeuds Connect:Direct distants», à la page 148 et «Format du fichier de propriétés du noeud Connect:Direct», à la page 603.

Connect:Direct utilise les informations de mappe de réseau pour déterminer la fin de ligne à utiliser.

- Si la destination d'un transfert est un Managed File Transfer Agent, cet agent effectue la conversion de fin de ligne.
- Si la destination d'un transfert est un noeud Connect:Direct , l'agent de pont Connect:Direct effectue la conversion de fin de ligne.

Référence associée

«Transfert de fichiers texte avec MFT», à la page 662

Le transfert de fichiers texte implique la conversion de la page de codes d'un fichier d'une page de codes à une autre. Le transfert de fichier texte implique également la conversion de caractères CRLF (retour chariot-saut de ligne) entre les systèmes. Cette rubrique récapitule le comportement de transfert de fichier texte de Managed File Transfer.

«Que faire si les transferts de texte vers ou depuis des noeuds Connect:Direct ne convertissent pas correctement les données», à la page 356

Lorsque vous transférez des fichiers en mode texte entre un agent MFT et un noeud Connect:Direct , la page de codes et la conversion de caractères de fin de ligne sont effectuées. Le transfert utilise les informations du système d'exploitation dans la mappe de réseau du noeud de pont Connect:Direct pour déterminer les caractères de fin de ligne d'un noeud distant. Si les informations de la mappe de réseau sont incorrectes, la conversion de caractères de fin de ligne peut être effectuée de manière incorrecte.

Transfert de fichiers vers ou depuis des agents de pont de protocole

Vous pouvez transférer des fichiers vers et depuis un serveur de fichiers FTP ou SFTP en dehors de votre réseau Managed File Transfer à l'aide d'un agent de pont de protocole.

Lorsque vous transférez des fichiers à l'aide du pont de protocole, le pont doit avoir le droit de lire le répertoire source ou de destination contenant les fichiers que vous souhaitez transférer. Par exemple, si

vous souhaitez transférer des fichiers à partir du répertoire `/home/fte/bridge` qui ne dispose que des droits d'exécution (`d -- x -- x -- x`), les transferts que vous tentez d'effectuer à partir de ce répertoire échouent avec le message d'erreur suivant:

```
BFGBR0032E: Attempt to read filename from the protocol file server has failed with server error 550
Failed to open file.
```

Lors du transfert de fichiers, les fichiers sont généralement écrits en tant que fichiers temporaires sur la destination et sont ensuite renommés lorsque le transfert est terminé. Toutefois, si la destination de transfert est un serveur de fichiers de protocole configuré en écriture limitée (les utilisateurs peuvent télécharger des fichiers sur le serveur de fichiers de protocole mais ne peuvent pas modifier ces fichiers téléchargés de quelque manière que ce soit ; les utilisateurs peuvent écrire une seule fois), les fichiers transférés sont écrits directement sur la destination. Cela signifie que si un problème se produit lors du transfert, les fichiers partiellement écrits restent sur le serveur de fichiers de protocole de destination et Managed File Transfer ne peut pas supprimer ou modifier ces fichiers. Dans ce cas, le transfert échoue.

Vérifiez que vous disposez d'un autre agent dans votre réseau Managed File Transfer en plus de l'agent de pont de protocole. L'agent de pont de protocole est un pont vers le serveur FTP ou SFTP uniquement et n'écrit pas les fichiers transférés sur le disque local. Si vous souhaitez transférer des fichiers vers ou depuis le serveur FTP ou SFTP, vous devez utiliser l'agent de pont de protocole comme destination ou source pour le transfert de fichiers (représentant le serveur FTP ou SFTP) et un autre agent standard comme source ou destination correspondante.

Demandes de transfert gérées nécessitant la création d'un nouveau répertoire sur un serveur de fichiers SFTP

Les agents de pont de protocole Managed File Transfer utilisent la bibliothèque JSch tierce pour communiquer avec les serveurs de fichiers à l'aide du protocole SFTP. Si l'agent de pont de protocole tente de transférer un fichier dans un répertoire qui n'existe pas sur un serveur de fichiers et que JSch ne peut pas effectuer l'opération SFTP demandée pour créer ce répertoire car l'utilisateur avec lequel l'agent de pont de protocole s'est connecté au serveur de fichiers ne dispose pas des droits appropriés, JSch envoie une exception à l'agent de pont de protocole. L'agent de pont de protocole marque ensuite le transfert géré comme ayant échoué et génère un message complémentaire. Si JSch a fourni plus d'informations sur l'échec, l'agent de pont de protocole inclut ces informations dans le message supplémentaire:

```
BFGTR0072E: Le transfert a échoué en raison de l'exception suivante :
BFGBR0119E: L'agent de pont n'a pas pu créer le répertoire nom_répertoire car le message de
l'exception JSch
```

V 9.0.0.9 Depuis la IBM MQ 9.0.0 Fix Pack 9, si l'exception JSch ne contient pas d'autres informations sur l'échec, l'agent de pont de protocole génère le message complémentaire suivant :

```
BFGTR0072E: Le transfert a échoué en raison de l'exception suivante :
BFGBR0209E: L'agent de pont n'est pas parvenu à créer le répertoire nom_répertoire
```

Concepts associés

«Le pont de protocole», à la page 219

Le pont de protocole permet à votre réseau Managed File Transfer (MFT) d'accéder aux fichiers stockés sur un serveur de fichiers en dehors de votre réseau MFT, dans votre domaine local ou à un emplacement distant. Ce serveur de fichiers peut utiliser les protocoles de réseau FTP, FTPS ou SFTP. Chaque serveur de fichiers a besoin d'au moins un agent dédié. L'agent dédié est appelé agent de pont de protocole. Un agent de pont peut interagir avec plusieurs serveurs de fichiers.

IBM i *Transfert de fichiers vers ou depuis des systèmes IBM i*

Si vous transférez des fichiers vers ou depuis des systèmes IBM i à l'aide de Managed File Transfer en mode texte et que vous souhaitez convertir les données des fichiers, prenez en compte les informations de cette rubrique.

Chaque fichier d'un système IBM i est balisé avec une valeur d'ID de jeu de caractères codés (CCSID) qui identifie le codage de données du fichier. Par exemple, un fichier contenant des données EBCDIC peut

avoir une valeur de CCSID de 037 et un fichier contenant des données ASCII peut avoir une valeur de CCSID de 819.

Pour les transferts en mode texte, Managed File Transfer convertit les données lorsqu'il existe des différences de codage de fichier entre les fichiers source et cible. Toutefois, Managed File Transfer ignore actuellement les balises CCSID associées aux fichiers sur les systèmes IBM i . A la place, il utilise la propriété de codage de fichier JVM des machines virtuelles Java exécutant l'agent source et l'agent de destination. La valeur par défaut de cette propriété est basée sur l'environnement local (mais vous pouvez la remplacer sur votre système IBM i à l'aide du fichier `SystemDefault.properties` décrit dans la section suivante:«[Modification de l'enregistrement file.encoding dans le fichier SystemDefault.properties](#)», à la page 667). Avec cette implémentation par défaut, un agent qui transfère des fichiers en mode texte est limité dans sa capacité à gérer des fichiers texte avec des codages de fichiers différents. Par exemple, vous ne pouvez pas utiliser le même agent pour transférer des fichiers contenant du texte EBCDIC et des fichiers contenant du texte ASCII sans arrêter et redémarrer l'agent avec la substitution de codage de fichier appropriée (c'est-à-dire EBCDIC ou ASCII) en place. Sur les systèmes IBM i V6R1 , vous pouvez vérifier la valeur de codage de fichier de la machine virtuelle Java qui exécute le travail d'agent à l'aide de la commande `WRKJVMJOB`, option 7 pour afficher les propriétés système En cours Java . (La commande `WRKJVMJOB` n'existe pas sur les systèmes IBM i V5R4 .)

Si vous prévoyez d'utiliser Managed File Transfer pour transférer des fichiers texte avec des codages de fichier différents, envisagez de créer plusieurs agents et plusieurs utilisateurs qui démarrent ces agents, de sorte que chaque codage unique ait un agent prêt et activé pour transférer ce type de données.

Par exemple, si vous souhaitez transférer un fichier contenant du texte EBCDIC avec la valeur de CCSID 037 d'un système IBM i (source) vers un autre système IBM i V6R1 (destination) où vous souhaitez convertir le contenu du fichier à la destination en texte ASCII avec la valeur de CCSID 819, procédez comme suit:

1. Sélectionnez un agent source avec un codage de fichier JVM Cp037.
2. Sélectionnez un agent de destination avec un codage de fichier JVM ISO8859_1.
3. Sélectionnez le transfert en mode texte et d'autres spécifications si nécessaire.

Modification de l'enregistrement `file.encoding` dans le fichier `SystemDefault.properties`

Pour activer une machine virtuelle Java exécutant un agent pour un codage particulier, procédez comme suit:

1. Déterminez quel utilisateur démarre l'agent qui s'exécute sur le système IBM i . Il s'agit de l'agent qui sert la demande de transfert de fichier Managed File Transfer .

Créez un fichier `SystemDefault.properties` dans le répertoire de base de cet utilisateur, si nécessaire. Par exemple, si vous démarrez l'agent, utilisez Qshell pour exécuter la commande suivante:

```
touch -C 819 /home/your_userID/SystemDefault.properties
```

2. A l'aide de Qshell, exécutez la commande `/qibm/proddata/mqm/bin/fteStopAgent` pour arrêter l'agent si nécessaire.
3. Mettez à jour le fichier `SystemDefault.properties` décrit à l'étape 1 pour vous assurer qu'il contient un enregistrement du type suivant:

```
file.encoding=java_encoding
```

où `java encoding` correspond au type de données contenues dans le fichier et correspond à une valeur `file.encoding` du tableau suivant: [File.encoding values](#) et [System i5 CCSID](#).

4. L'utilisateur identifié à l'étape 1 doit effectuer les étapes suivantes:

- a. Sur IBM i V5R4 uniquement: ajoutez la variable d'environnement QIBM_PASE_DESCRIPTOR_STDIO (portée *JOB) à 'B' si vous utilisez le codage de fichier EBCDIC ou à 'T' si vous utilisez le codage ASCII. Exemple :

```
ADDENVVAR ENVVAR('QIBM_PASE_DESCRIPTOR_STDIO') VALUE('B') REPLACE(*YES)
```

- b. Si Qshell est actif, appuyez sur **F3=Exit** pour arrêter Qshell.
- c. Démarrez Qshell et exécutez la commande **/qibm/proddata/mqm/bin/fteStartAgent** comme il convient pour redémarrer l'agent.

Lorsque le codage de fichier de la machine virtuelle Java exécutant l'agent a été modifié, le journal de l'agent est écrit avec ce codage. Si vous souhaitez lire le contenu du journal de l'agent, vous devez utiliser un afficheur activé pour ce codage.

Utilisation d'une définition de transfert pour la conversion de données

Vous pouvez également convertir des données lorsque des fichiers sont transférés en créant une définition de transfert qui spécifie le codage de fichier ou en utilisant les paramètres **-sce** et **-dce** de la commande **fteCreateTransfer**. Si vous utilisez ces paramètres lorsque la destination est un système IBM i, cela peut générer des fichiers dont les balises CCSID sont incorrectes. Pour cette raison, l'approche recommandée pour contrôler la conversion de données avec les fichiers qui se trouvent sur les systèmes IBM i consiste à utiliser `SystemDefault.properties` comme décrit dans la section précédente.

Limitation du pont de protocole

Sous IBM i, vous ne pouvez pas transférer des fichiers EBCDIC vers ou depuis un serveur SFTP à l'aide d'un agent de pont de protocole.

Référence associée

«Instructions pour le transfert de fichiers», à la page 648

Selon le système d'exploitation depuis et vers lequel vous effectuez le transfert et selon que vous effectuez le transfert en mode binaire ou texte, il existe des instructions sur le comportement à attendre.

«Transfert des fichiers sauvegarde situés dans QSYS.LIB sous IBM i», à la page 671

Managed File Transfer prend en charge le transfert des fichiers sauvegarde situés dans QSYS.LIB entre deux systèmes IBM i. Tenez compte des informations suivantes lorsque vous demandez des transferts de fichiers sauvegarde.

Information associée

[Installation du serveur IBM MQ sous IBM i](#)

IBM i *Transfert de fichiers physiques situés dans QSYS.LIB sous IBM i*

Managed File Transfer prend en charge le transfert des membres de fichier physique dans QSYS.LIB entre deux systèmes IBM i. Tenez compte des informations suivantes lorsque vous demandez des transferts de fichiers de membres de fichiers physiques.

Un membre de fichier physique sous IBM i se trouve dans un fichier physique qui, à son tour, se trouve dans une bibliothèque sous IBM i. Une bibliothèque peut être l'une des bibliothèques standard qui sont livrées avec le système d'exploitation (par exemple, QSYS ou QGPL) ou il peut s'agir d'une bibliothèque que vous avez créée.

Fichiers physiques dans QSYS.LIB est identifié de deux manières différentes sur IBM i. Lorsque vous exécutez des commandes CL sur une ligne de commande IBM i, utilisez la syntaxe de dénomination suivante:

```
FILE(library name/file name) MBR(member name)
```

Par exemple, un membre de fichier physique appelé MYMBR se trouve dans un fichier appelé MYFILE dans une bibliothèque appelée SOMELIB et identifié comme FILE (SOMELIB/MYFILE) MBR (MYMBR). Vous pouvez également identifier le même membre de fichier physique en spécifiant un nom de chemin de type UNIX qui respecte la convention de dénomination du Integrated File System (IFS). A l'aide de la convention de dénomination IFS, MYMBR dans MYFILE dans SOMELIB possède le nom de chemin suivant:

```
/QSYS.LIB/SOMELIB.LIB/MYFILE.FILE/MYMBR.MBR
```

Pour plus d'informations, voir [Noms de chemin dans QSYS.LIB système de fichiers](#).

Managed File Transfer on IBM i reconnaît la convention de dénomination IFS mais ne prend pas en charge la syntaxe utilisée par les commandes CL. Les exemples suivants illustrent des noms de chemin valides et non valides pour MFT. L'exemple suivant est un nom de chemin valide pour un membre de fichier physique:

```
/QSYS.LIB/SOMELIB.LIB/MYFILE.FILE/MYMBR.MBR
```

Cet exemple suppose que MYFILE est un fichier physique de la bibliothèque SOMELIB et qu'il contient un membre appelé MYMBR.

Les exemples suivants sont des noms de chemin non valides pour les transferts de membres de fichiers physiques:

- /QSYS.LIB/SOMELIB.LIB/MYFILE.FILE (.FILE suppose une commande SAVF et non un fichier physique. Si MYFILE est un fichier physique, le transfert échoue avec une erreur de type de fichier non valide)
- /QSYS.LIB/MYLIB.LIB/ (les noms de fichier physique et de membre sont requis)
- /QSYS.LIB/SOMELIB.LIB/MYFILE.FILE/MYMBR (le nom de membre doit contenir une extension .MBR)
- /QSYS.LIB/SOMELIB.LIB/MYFILE/MYMBR.MBR (l'extension du nom de fichier physique doit être .FILE)

Transfert de plusieurs membres de fichier physique à partir d'un fichier physique dans une seule demande de transfert

Managed File Transfer on IBM i prend en charge le transfert de plusieurs membres de fichier physique à partir d'un seul fichier physique en tant que demande de transfert unique. Vous pouvez spécifier un nom de chemin approprié qui inclut des caractères génériques, comme illustré dans les exemples suivants:

- ABCLIB contient un fichier physique MYFILE avec plusieurs membres. Pour transférer tous ces membres dans une seule demande, indiquez le nom de chemin suivant: /QSYS.LIB/ABCLIB.LIB/MYFILE.FILE/* .MBR
- XYZLIB contient un fichier physique MYFILE dont les noms de membre diffèrent d'un seul caractère, à savoir: TEST1.MBR, TEST2.MBR, TEST3.MBR, etc. Pour transférer tous ces membres dans une seule demande, indiquez le nom de chemin suivant: /QSYS.LIB/XYZLIB.LIB/MYFILE.FILE/TEST? .MBR.

Les types de demande de transfert suivants ne sont pas pris en charge pour le transfert de plusieurs membres de fichier physique et génèrent une erreur:

- /QSYS.LIB/MYLIB.LIB/*.*
- /QSYS.LIB/MYLIB.LIB/*
- /QSYS.LIB/MYLIB.LIB/* .FILE/MYMBR.MBR
- /QSYS.LIB/MYLIB.LIB/MYFILE* .FILE/* .MBR (il n'y a pas de prise en charge des caractères génériques sur les noms de fichier, uniquement sur les noms de membre)
- /QSYS.LIB/MYLIB.LIB/* .FILE/* .MBR

- /QSYS.LIB/MYLIB.LIB/MYFILE.FILE (.FILE suppose que SAVF n'est pas un fichier physique, donc si MYFILE est un fichier physique, le transfert échoue avec une erreur de type de fichier non valide)

Transfert de membres de fichiers physiques vers et depuis des systèmes nonIBM i

MFT prend en charge le transfert de membres de fichiers physiques vers et depuis des systèmes nonIBM i, tels que UNIX, Linux, and Windows. Tous les transferts doivent être effectués en mode texte. Les exemples suivants illustrent certaines des demandes **fteCreateTransfer** prises en charge lors de l'utilisation de systèmes nonIBM i :

- Cette commande transfère le membre de fichier physique FILE (FROMIBMI/FILE1) MBR (FILE1) sous IBM i vers le fichier texte /home/qfte/fromibmi/linux.mbr.txt sous Linux:

```
fteCreateTransfer -da linux -dm QM1 -sa ibmi -sm QM1 -t text -df /home/qfte/fromibmi/
linux.mbr.txt /qsys.lib/fromibmi.lib/file1.file/file1.mbr
```

- Cette commande transfère le membre de fichier physique FILE (FROMIBMI/FILE1) MBR (FILE1) sous IBM i vers le fichier texte C:\FTE\fromibmi\windows.mbr.txt sous Windows:

```
fteCreateTransfer -da windows -dm QM1 -sa ibmi -sm QM1 -t text -df
C:\FTE\fromibmi\windows.mbr.txt /qsys.lib/fromibmi.lib/file1.file/file1.mbr
```

- Cette commande transfère le fichier texte C:\FTE\toibmi\file.txt sous Windows vers le membre de fichier physique FILE (TOIBMI/EXISTS) MBR (WINDOWS) sous IBM i:

```
fteCreateTransfer -da ibmi -dm QM1 -sa windows -sm QM1 -t text -df /qsys.lib/toibmi.lib/
exists.file/windows.mbr C:\FTE\toibmi\file.txt
```

Les commandes suivantes sont des exemples de transferts de membres de fichiers physiques non valides avec des systèmes nonIBM i :

- Cette commande échoue car le fichier source sous Windows a une extension de fichier .txt mais un répertoire de destination .file a été spécifié. Lors du transfert à l'aide du paramètre de répertoire de destination pour spécifier un fichier physique de destination, l'extension de fichier source doit être un fichier .mbr, par exemple, C:\FTE\toibmi\file.mbr

```
fteCreateTransfer -da ibmi -dm QM1 -sa windows -sm QM1 -t text -dd /qsys.lib/toibmi.lib/
windows.file C:\FTE\toibmi\file.txt
```

- Le mode de transfert par défaut est binaire et le mode texte doit être spécifié lors du transfert des membres de fichier physique.

```
fteCreateTransfer -da windows -dm QM1 -sa ibmi -sm QM1 -df C:\FTE\fromibmi\file.bin /qsys.lib/
fromibmi.lib/file1.file/file1.mbr
```

MFT prend en charge le transfert des membres de fichier physique qui se trouvent dans QSYS.LIB ne prend pas en charge le transfert des membres de fichier physique source qui se trouvent dans la bibliothèque QSYS.QSYS.LIB. Les transferts de fichiers dans le système de fichiers QDLS sont pris en charge à l'aide des exemples d'exit utilisateur fournis. Vous pouvez utiliser les exemples d'exit utilisateur fournis dans MFT pour les tâches suivantes:

- Transférer les fichiers dans le système de fichiers QDLS.
- Transférez automatiquement les membres de fichiers physiques d'une bibliothèque IBM i de la même manière qu'un moniteur de fichiers MFT.
- Supprimer un objet de fichier vide lorsque le membre de fichier source est supprimé dans le cadre du transfert.

Pour plus d'informations, voir la section «Exemples d'exit utilisateur MFT on IBM i», à la page 268.

Référence associée

«Instructions pour le transfert de fichiers», à la page 648

Selon le système d'exploitation depuis et vers lequel vous effectuez le transfert et selon que vous effectuez le transfert en mode binaire ou texte, il existe des instructions sur le comportement à attendre.

«Transfert de fichiers vers ou depuis des systèmes IBM i», à la page 666

Si vous transférez des fichiers vers ou depuis des systèmes IBM i à l'aide de Managed File Transfer en mode texte et que vous souhaitez convertir les données des fichiers, prenez en compte les informations de cette rubrique.

IBM i **Transfert des fichiers sauvegarde situés dans QSYS.LIB sous IBM i**

Managed File Transfer prend en charge le transfert des fichiers sauvegarde situés dans QSYS.LIB entre deux systèmes IBM i. Tenez compte des informations suivantes lorsque vous demandez des transferts de fichiers sauvegarde.

Un fichier sauvegarde sous IBM i se trouve dans une bibliothèque sous IBM i. Une bibliothèque peut être l'une des bibliothèques standard fournies avec le système d'exploitation, par exemple QSYS ou QGPL, ou une bibliothèque créée par l'utilisateur. Sauvegardez les fichiers dans QSYS.LIB est identifié de deux manières différentes sur IBM i. Lorsque vous utilisez des commandes CL sur une ligne de commande IBM i, la syntaxe de dénomination utilisée est la suivante:

```
FILE(library name/file name)
```

Par exemple, un fichier sauvegarde appelé MYSAVF se trouve dans une bibliothèque appelée SOMELIB et est identifié comme FILE (SOMELIB/MYSAVF).

Vous pouvez également identifier le même fichier sauvegarde en spécifiant un nom de chemin de type UNIX qui respecte la convention de dénomination du Integrated File System (IFS). Voir [Noms de chemin dans QSYS.LIB système de fichiers](#) pour plus d'informations. A l'aide de la convention de dénomination IFS, MYSAVF dans SOMELIB possède le nom de chemin suivant:

```
/QSYS.LIB/SOMELIB.LIB/MYSAVF.FILE
```

Managed File Transfer on IBM i reconnaît la convention de dénomination IFS mais ne prend pas en charge la syntaxe utilisée par les commandes CL. Les exemples suivants illustrent des noms de chemin valides et non valides pour Managed File Transfer.

Voici quelques exemples de noms de chemin valides pour les transferts de fichiers sauvegarde:

- /QSYS.LIB/SOMELIB.LIB/MYSAVF.FILE (en supposant que le fichier sauvegarde MYSAVF se trouve dans la bibliothèque SOMELIB)
- /QSYS.LIB/MYSAVF.FILE (en supposant que MYSAVF se trouve dans la bibliothèque QSYS)

Voici quelques exemples de noms de chemin non valides pour les transferts de fichiers de sauvegarde:

- SOMELIB.LIB/MYSAVF.FILE (le nom de chemin doit commencer par /QSYS.LIB)
- /QSYS.LIB/MYLIB.LIB (le chemin doit se terminer par un nom de fichier sauvegarde et non par un nom de bibliothèque)
- /QSYS.LIB/MYLIB.LIB/ (le nom du fichier sauvegarde est obligatoire)
- /QSYS.LIB/SOMELIB.LIB/MYSAVF (Le nom du fichier sauvegarde doit avoir une extension .FILE dans le nom)
- /QSYS.LIB/SOMELIB.LIB/MYSAVF.SAVF (l'extension de nom de fichier de sauvegarde doit être .FILE)

Transfert de plusieurs fichiers sauvegarde à partir d'une bibliothèque dans une même demande de transfert

Managed File Transfer on IBM i prend en charge le transfert de plusieurs fichiers sauvegarde à partir d'une bibliothèque en tant que demande de transfert unique. Vous pouvez spécifier un nom de chemin approprié qui inclut des caractères génériques, comme illustré dans les exemples suivants:

- ABCLIB contient de nombreux fichiers sauvegarde. Pour transférer tous ces fichiers en une seule demande, indiquez le nom de chemin suivant:

```
/QSYS.LIB/ABCLIB.LIB/*.FILE
```

- XYZLIB contient plusieurs fichiers sauvegarde dont les noms diffèrent d'un seul caractère, à savoir: TEST1.FILE, TEST2.FILE, TEST3.FILE, etc. Pour transférer tous ces fichiers dans une seule demande, indiquez le nom de chemin suivant:

```
/QSYS.LIB/XYZLIB.LIB/TEST?.FILE
```

Les types de demande de transfert suivants ne sont pas pris en charge pour le transfert de plusieurs fichiers sauvegarde et génèrent une erreur:

- /QSYS.LIB/MYLIB.LIB/*.*

- /QSYS.LIB/MYLIB.LIB/*

Managed File Transfer prend en charge le transfert des fichiers sauvegarde qui se trouvent dans QSYS.LIB, mais le transfert d'autres types de fichiers qui se trouvent dans QSYS.LIB n'est pas pris en charge. Toutefois, Managed File Transfer fournit des exemples qui utilisent la prise en charge des fichiers de sauvegarde et des tâches fteAnt prédéfinies pour montrer comment une bibliothèque complète, un fichier physique source ou un fichier de base de données peut être transféré entre deux systèmes IBM i. Pour plus d'informations sur la personnalisation et l'utilisation de ces exemples, voir [«Initiation à l'utilisation de scripts Ant avec MFT»](#), à la page 259.

Référence associée

«Instructions pour le transfert de fichiers», à la page 648

Selon le système d'exploitation depuis et vers lequel vous effectuez le transfert et selon que vous effectuez le transfert en mode binaire ou texte, il existe des instructions sur le comportement à attendre.

«Transfert de fichiers vers ou depuis des systèmes IBM i», à la page 666

Si vous transférez des fichiers vers ou depuis des systèmes IBM i à l'aide de Managed File Transfer en mode texte et que vous souhaitez convertir les données des fichiers, prenez en compte les informations de cette rubrique.

Transfert de groupes de données de génération (GDG)

Managed File Transfer prend en charge les groupes de données de génération (GDG) pour les fichiers source et de destination sur z/OS. Les noms d'ensemble de fichiers absolus et relatifs sont pris en charge. Lorsque vous écrivez dans une nouvelle génération, l'ensemble de fichiers de base doit exister.

Remarque : Lors de la création d'une entrée GDG dans un environnement de traitement par lots à l'aide de BASEGDG (+n), elle ne peut pas être référencée ultérieurement dans le même travail en utilisant le même numéro de génération positif. La conservation des mêmes numéros d'entrée d'ensemble de fichiers entre les étapes d'un travail est une fonction de JCL et n'est pas disponible pour les fonctions d'utilitaire qui mettent à jour l'ensemble de fichiers à l'aide de l'allocation dynamique. Par conséquent, un travail qui crée une nouvelle génération à l'aide de BASEGDG (+ 1) trouvera l'ensemble de fichiers mis à jour dès que le transfert aura abouti et devra alors faire référence au même ensemble de données que BASEGDG (0).

Exemples d'ensemble de fichiers

Les exemples suivants illustrent la commande **fteCreateTransfer** utilisant des groupes de données de génération (GDG). Dans les exemples, le nom BASEGDG fait référence à un nom d'ensemble de fichiers de base existant. Le nom DSET fait référence à un fichier séquentiel à créer. Le nom /u/user/file.dat fait référence au nom d'un fichier de données source.

Cette commande copie file.dat dans une nouvelle génération dans BASEGDG. Le nom absolu de la nouvelle génération est indiqué dans le journal de transfert:

```
fteCreateTransfer -sa A1 -da A2 -ds "//BASEGDG(+1)" /u/user/file.dat
```

Cette commande copie file.dat dans la génération avec le nom absolu spécifié dans BASEGDG:

```
fteCreateTransfer -sa A1 -da A2 -ds "//BASEGDG.G0009V00" /u/user/file.dat
```

Cette commande copie la génération la plus récente dans BASEGDG vers DSET. Le nom absolu de la génération est indiqué dans le journal de transfert:

```
fteCreateTransfer -sa A1 -da A2 -ds "//DSET" "//BASEGDG(0)"
```

Cette commande copie la génération la plus récente de BASEGDG vers DSET. Le nom absolu de la génération est indiqué dans le journal de transfert:

```
fteCreateTransfer -sa A1 -da A2 -ds "//DSET" "//BASEGDG(-1)"
```

Référence associée

«Instructions pour le transfert de fichiers», à la page 648

Selon le système d'exploitation depuis et vers lequel vous effectuez le transfert et selon que vous effectuez le transfert en mode binaire ou texte, il existe des instructions sur le comportement à attendre.

«fteCreateTransfer: démarrer un nouveau transfert de fichier», à la page 447

La commande **fteCreateTransfer** crée et démarre un nouveau transfert de fichier à partir de la ligne de commande. Cette commande peut démarrer un transfert de fichier immédiatement, planifier un transfert de fichier à une heure et une date ultérieures, répéter un transfert planifié une ou plusieurs fois et déclencher un transfert de fichiers sous certaines conditions.

«Transfert entre fichiers sous z/OS», à la page 651

Vous pouvez effectuer un transfert entre des fichiers z/OS à l'aide de Managed File Transfer. Examinez attentivement le comportement suivant pour vous assurer que vos fichiers sont transférés correctement.

Utilisation de caractères génériques avec MFT

Vous pouvez utiliser des caractères génériques lorsque vous spécifiez des noms de fichier source et des chemins de fichier source pour les transferts de fichiers. Cela vous permet de sélectionner plusieurs fichiers simultanément.

Multiplateformes

Vous pouvez utiliser les caractères génériques suivants sous Multiplateformes:

?

Utilisez le point d'interrogation (?) pour représenter exactement un caractère. Tous les autres caractères indiqués sont obligatoires dans les noms de fichier correspondants.

Par exemple, ab?.jpg correspond aux fichiers abcd.jpg, abed.jpg et abfd.jpg.

*

Utilisez l'astérisque (*) pour représenter zéro ou plusieurs caractères.

Par exemple, *.txt correspond aux fichiers abc.txt et x.txt, mais pas à newtxt car le point (.) dans les noms de fichier est un caractère obligatoire.

Le modèle *txt correspond aux fichiers abc.txt, x.txt et newtxt.

Vous devez placer l'astérisque (*) entre guillemets. Si vous ne le faites pas, le caractère sera interprété par l'interpréteur de commandes et peut entraîner l'échec de la commande.

  Sous UNIX and Linux, l'utilisation de l'astérisque (*) n'inclut pas les fichiers pseudo-masqués, par exemple .bashrc.

Si le système d'exploitation est insensible à la casse pour les noms de fichier et de chemin d'accès, par exemple Windows, la correspondance de modèle est insensible à la casse. Vous pouvez utiliser des caractères génériques pour spécifier des noms de fichier uniquement: vous ne pouvez pas utiliser de caractères génériques dans les noms de répertoire.

Agent de pont de protocole

Si vous utilisez un agent de pont de protocole pour transférer des fichiers à partir d'un serveur de fichiers FTP, FTPS ou SFTP, la mise en correspondance des caractères génériques est sensible à la casse, quelle que soit la plateforme sur laquelle le serveur de fichiers s'exécute.

Connect:Direct pont

Lorsque la source d'un transfert est un agent de pont Connect:Direct qui demande des fichiers à un noeud Connect:Direct, les caractères génériques ne sont pas pris en charge.

IBM i



Vous pouvez utiliser les caractères génériques suivants sur les plateformes IBM i :

?

Utilisez le point d'interrogation (?) pour représenter exactement un caractère. Tous les autres caractères indiqués sont obligatoires dans les noms de fichier correspondants.

Par exemple, ab?.jpg correspond aux fichiers abcd.jpg, abed.jpg et abfd.jpg.

*

Utilisez l'astérisque (*) pour représenter zéro ou plusieurs caractères.

Par exemple, *.txt correspond aux fichiers abc.txt et x.txt.

Le modèle *txt correspond aux fichiers abc.txt, x.txt et newtxt car le point (.) dans le modèle est un caractère obligatoire.

Pour plus d'informations sur l'utilisation de caractères génériques avec les transferts de fichiers sauvegarde, voir [Transfert de fichiers sauvegarde résidant dans QSYS.LIB sur les systèmes IBM i](#).

z/OS



Pour les systèmes z/OS, les règles de caractères génériques pour Managed File Transfer respectent les conventions de caractères génériques ISPF standard en général. Il existe des règles spécifiques pour les fichiers séquentiels et partitionnés comme suit:

Ensembles de données séquentielles



Lorsque vous référencez des fichiers séquentiels, vous pouvez utiliser des qualificatifs de nom de fichier contenant des astérisques (*) et des signes de pourcentage (%) comme suit:

*

Utilisez un astérisque (*) unique pour représenter au moins un qualificateur. Un astérisque unique dans un qualificateur représente zéro ou plusieurs caractères.

**

Utilisez des astérisques doubles (**) pour représenter zéro ou plusieurs qualificatifs. Vous ne pouvez pas utiliser un double astérisque dans un qualificateur.

%

Utilisez un signe de pourcentage unique (%) pour représenter un caractère alphanumérique unique ou un caractère de langue nationale.

%%

Utilisez entre un et huit signes de pourcentage pour représenter zéro ou plusieurs caractères.

Ensemble de données partitionnées

 Lorsque vous référencez des fichiers partitionnés, vous pouvez spécifier des caractères génériques uniquement pour les noms de membre. Vous pouvez utiliser des qualificateurs de nom de fichier contenant des astérisques (*), des traits de soulignement (_) et des points d'interrogation (?) comme suit:

*

Utilisez l'astérisque (*) pour représenter zéro ou plusieurs caractères.

_

Utilisez le trait de soulignement (_) pour représenter exactement un caractère.

?

Utilisez le point d'interrogation (?) pour représenter exactement un caractère. Le point d'interrogation est une alternative au caractère de soulignement et est fourni en tant qu'ajout aux conventions ISPF.

Répertoires

Par défaut, si vous créez un transfert de fichiers avec un masque de caractères génériques qui correspond à des sous-répertoires, les sous-répertoires ne sont pas transférés. Vous pouvez spécifier le paramètre **-r** dans la commande `fteCreateTransfer` pour inclure les sous-répertoires qui correspondent au masque de caractères génériques. Lorsque vous transférez un sous-répertoire, l'intégralité du contenu et de la structure du sous-répertoire sont transférés, y compris tous ses fichiers, sous-répertoires et fichiers masqués.

Par exemple, si vous disposez d'un répertoire appelé abc, il existe une différence de comportement entre la spécification d'un chemin de fichier source `/opt/abc` et `/opt/abc/*`. Dans le cas de `/opt/abc` car le répertoire est transféré, un répertoire appelé abc est créé à la destination et tout le contenu du fichier est transféré. Dans le cas de `/opt/abc/*`, le contenu de abc est transféré dans le chemin de destination.

Fichiers masqués

Les caractères génériques ne correspondent pas aux fichiers masqués sauf sur les plateformes de type UNIX lorsque le masque de caractères génériques commence par un point (.). Par exemple: `/opt/.*` transfère tous les fichiers masqués dans le répertoire `opt`.

Sous Windows, si vous souhaitez transférer un fichier masqué, indiquez le nom de fichier exactement ou transférez le répertoire contenant le fichier masqué.

Liens symboliques

Les liens symboliques sont un type de fichier qui contient un pointeur vers un autre fichier ou répertoire et sont appelés raccourcis sous Windows. Vous pouvez faire correspondre des fichiers de liens symboliques avec des caractères génériques. Toutefois, lorsqu'un fichier de destination est créé à partir d'une source qui est un lien symbolique, le fichier de destination devient un lien matériel (c'est-à-dire un fichier standard). Vous ne pouvez pas transférer des liens symboliques vers des répertoires car cela pourrait potentiellement créer un chemin récursif.

Transfert de fichiers avec des caractères génériques dans leurs noms de fichier

Vous pouvez transférer un fichier si le nom de fichier lui-même contient un caractère générique. Si vous spécifiez exactement ce nom de fichier, seul ce fichier est transféré, et non l'ensemble de fichiers correspondant au caractère générique.

Par exemple, si vous disposez d'un fichier appelé `/opt/abc*.txt` et que vous créez un transfert de fichier pour `/opt/abc*.txt`, le seul fichier transféré est `/opt/abc*.txt`. Mais si vous créez un transfert de fichier pour `/opt/ab*.txt`, tous les fichiers correspondant au modèle `/opt/ab*.txt` sont transférés, y compris le fichier `/opt/abc*.txt`.

Transfert de chemins de répertoire contenant des caractères génériques

Placez tout chemin de répertoire incluant un caractère générique entre guillemets (") ou des guillemets simples (') pour éviter l'extension de l'interpréteur de commandes. L'extension de l'interpréteur de commandes se produit lorsque le système d'exploitation développe le caractère générique avant qu'il ne soit transmis à la commande Managed File Transfer, ce qui peut entraîner un comportement inattendu.

Par exemple, si vous exécutez la commande **fteCreateTransfer** suivante avec le paramètre **-gt** sous UNIX, où `${...}` est une substitution de variable à partir d'un moniteur de ressources:

```
fteCreateTransfer -p QM_VENUS -sa AGT.QM_JUPITER -sm QM_JUPITER -da AGT.QM_NEPTUNE -dm QM_NEPTUNE -r -sd delete -t binary -de overwrite -jn MONTASK -gt /home/ftadmin/bin/TransferTask.xml -df "${FilePath}" "${FilePath}"
```

le shell analyse `${FilePath}` et ne le transmet pas à la commande. La solution consiste à placer `${FilePath}` entre guillemets, c'est-à-dire `"${FilePath}"`.

Le transfert est signalé comme ayant abouti même si le caractère générique correspond à zéro fichier

Si vous tentez de transférer un fichier qui n'existe pas, Managed File Transfer traite cette tentative comme un échec de transfert. Si vous indiquez explicitement un nom de fichier (par exemple, `/a/missing/filename.txt`) et que MFT ne parvient pas à trouver ce fichier, le message d'erreur suivant est consigné dans le journal:

```
BFGI00001E: File "/a/missing/filename.txt" does not exist
```

Dans le cadre de ce processus, l'agent source, qui n'a pas pu trouver le fichier, informe l'agent cible que ce transfert de fichier a été annulé (car l'agent source ne trouve pas le fichier source à lire). Si vous aviez prévu de déclencher un exit après le transfert à ce stade, l'agent de destination déclenche son `DestinationTransferEndExit` avec un `FileExitResultCode` de `CANCEL_FILE` pour ce nom de fichier.

Toutefois, si vous tentez de transférer un caractère générique (par exemple, `/a/missing/*.txt`) et que l'agent source ne trouve aucun fichier correspondant à ce caractère générique, MFT signale que le transfert a abouti. En effet, techniquement, l'agent source a été invité à transférer 0 fichier. Le message d'erreur suivant est consigné dans le journal:

```
The transfer request has successfully completed, although no files were transferred.
```

Dans cet exemple, l'agent de destination n'ayant jamais été impliqué dans le transfert, son exit n'est pas appelé.

Référence associée

«Instructions pour le transfert de fichiers», à la page 648

Selon le système d'exploitation depuis et vers lequel vous effectuez le transfert et selon que vous effectuez le transfert en mode binaire ou texte, il existe des instructions sur le comportement à attendre.

«fteCreateTransfer: démarrer un nouveau transfert de fichier», à la page 447

La commande **fteCreateTransfer** crée et démarre un nouveau transfert de fichier à partir de la ligne de commande. Cette commande peut démarrer un transfert de fichier immédiatement, planifier un

transfert de fichier à une heure et une date ultérieures, répéter un transfert planifié une ou plusieurs fois et déclencher un transfert de fichiers sous certaines conditions.

Transfert entre deux topologies MFT différentes

Les agents Managed File Transfer (MFT) peuvent uniquement effectuer des transferts gérés entre d'autres agents dans la même topologie. Toutefois, si vous disposez de plusieurs topologies, il peut être utile de transférer des données entre elles. Le texte suivant fournit des conseils de haut niveau sur la façon de procéder.

Voici un diagramme qui montre deux topologies différentes:

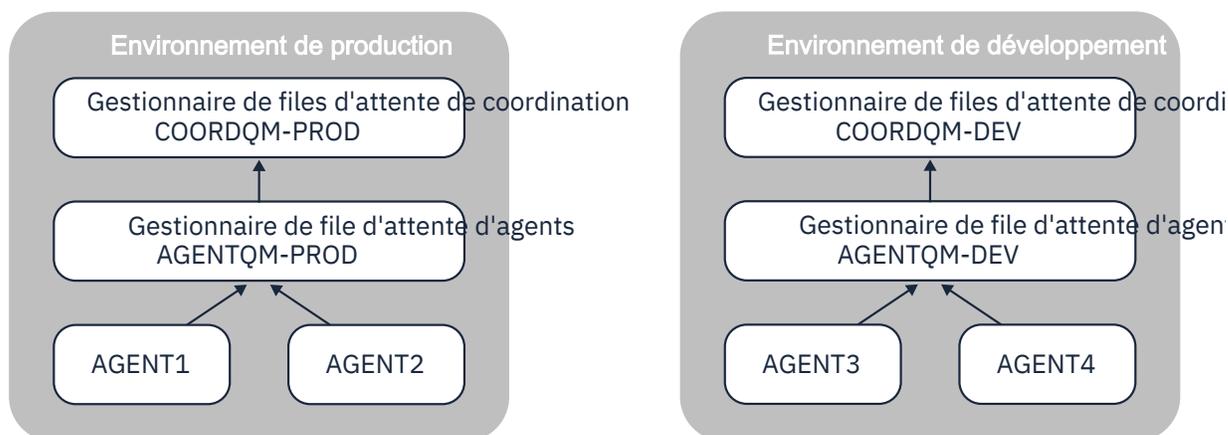


Figure 14. AGENT1 et AGENT2 font partie d'une topologie dans l'environnement de production, et AGENT3 et AGENT4 font partie de la topologie de l'environnement de développement.

La topologie de production est distincte de la topologie de développement. Cela signifie que les agents en production ne peuvent pas participer directement aux transferts gérés avec les agents dans l'environnement de développement (par exemple, AGENT2 ne peut pas effectuer de transfert géré vers AGENT3). Pour transférer des données entre les environnements, vous pouvez utiliser un système de fichiers partagé ou des transferts de fichier à message et de message à fichier.

Transfert de données à l'aide d'un système de fichiers partagé

Dans cette solution, les agents des deux topologies ont accès au même système de fichiers partagé.

Un agent dans une topologie agit en tant qu'agent de destination pour un transfert géré et écrit un fichier dans un emplacement connu sur le système de fichiers. Un autre agent de la deuxième topologie utilise un moniteur de ressources ou un transfert planifié pour détecter lorsqu'un fichier apparaît à cet emplacement, puis le traite.

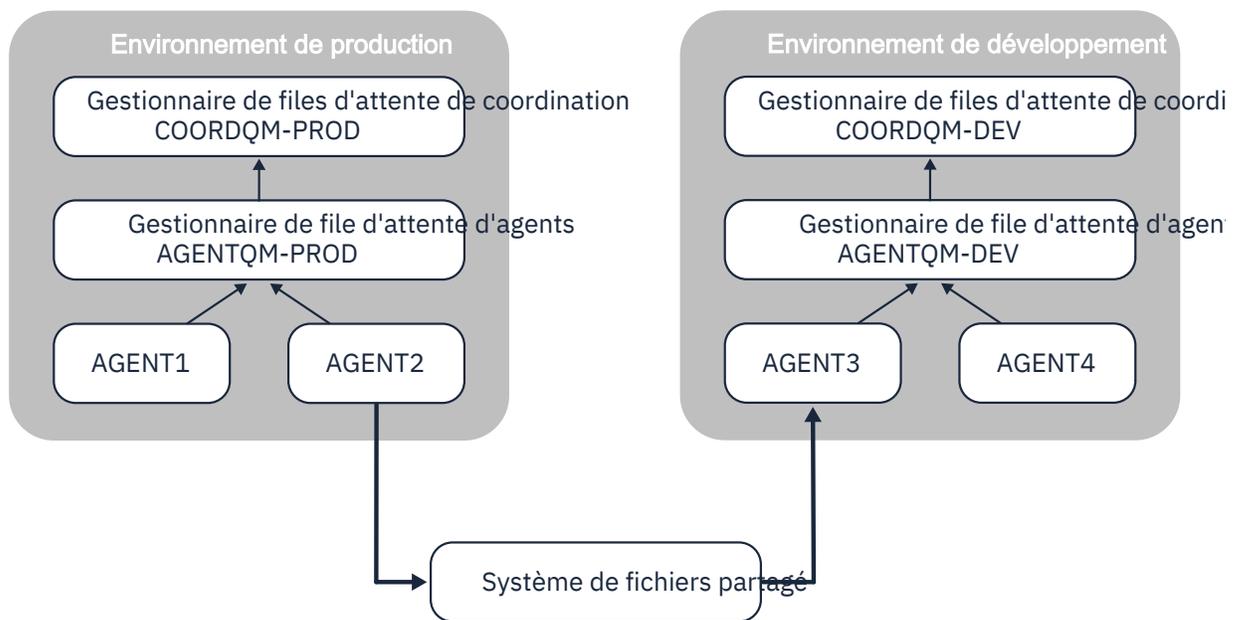


Figure 15. Ici, AGENT2 est l'agent de destination pour un transfert géré s'exécutant dans la topologie de l'environnement de production et écrit un fichier dans un emplacement sur le système de fichiers partagé. Un moniteur de ressources s'exécutant dans AGENT3 interroge cet emplacement. Lorsqu'il détecte le fichier écrit par AGENT2, il soumet une demande de transfert géré à AGENT3 pour le traiter et l'intégrer à la topologie de l'environnement de développement.

Notez que le système de fichiers partagé doit être fiable afin de s'assurer que les données ne sont pas perdues.

Transfert de données à l'aide de transferts de message à fichier et de fichier à message

Une autre approche consiste à utiliser un gestionnaire de files d'attente de passerelle entre les deux topologies. Ce gestionnaire de files d'attente est connecté aux gestionnaires de files d'attente d'agent dans les topologies à l'aide des canaux émetteur et récepteur, pour permettre la transmission des données entre les deux.

Un agent de l'une des topologies effectue un transfert de fichier à message, pour écrire des données dans une file d'attente éloignée. Le message est ensuite acheminé via le gestionnaire de files d'attente de passerelle vers une file d'attente locale sur un gestionnaire de files d'attente de l'autre topologie. Un agent de cette topologie effectue ensuite un transfert de message à fichier pour obtenir le message et le traiter.

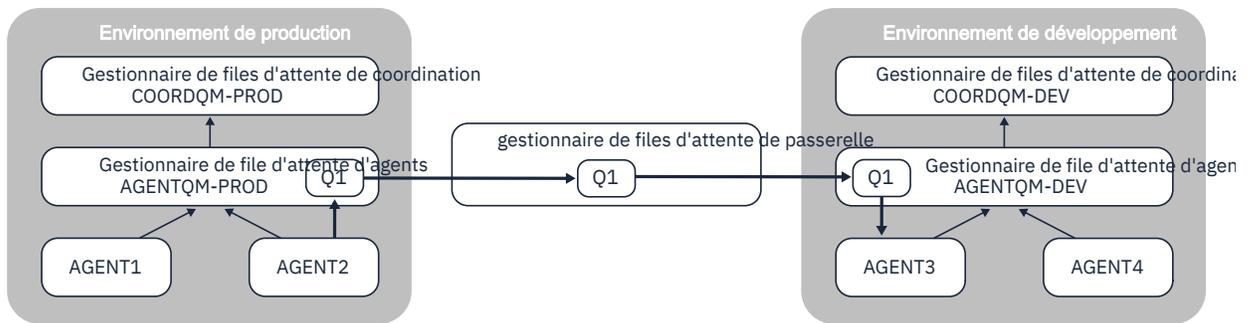


Figure 16. Ici, AGENT2 est connecté à son gestionnaire de files d'attente d'agent AGENTQM-PROD et effectue un transfert de fichier à message pour écrire un message dans une file d'attente appelée Q1. Q1 est une file d'attente éloignée. Par conséquent, le message est acheminé via le gestionnaire de files d'attente de passerelle et les canaux émetteur et récepteur vers la file d'attente locale Q1 sur le gestionnaire de files d'attente AGENTQM-DEV. AGENT3 effectue ensuite un transfert de message à fichier pour obtenir le message et l'insérer dans la topologie de l'environnement de développement.

Cette solution utilise la mise en réseau IBM MQ standard pour transférer des messages d'une topologie à une autre via le gestionnaire de files d'attente de passerelle. Cela signifie que si un canal entre le gestionnaire de files d'attente de passerelle et l'un des gestionnaires de files d'attente d'agent n'est pas disponible pour une raison quelconque, les messages risquent d'être bloqués et de ne pas arriver dans la file d'attente de destination. Dans ce cas, vous devez vérifier les canaux pour vous assurer qu'ils sont tous en cours d'exécution.

Référence associée

«Instructions pour le transfert de fichiers», à la page 648

Selon le système d'exploitation depuis et vers lequel vous effectuez le transfert et selon que vous effectuez le transfert en mode binaire ou texte, il existe des instructions sur le comportement à attendre.

Expressions régulières utilisées par MFT

Managed File Transfer utilise des expressions régulières dans un certain nombre de scénarios. Par exemple, des expressions régulières sont utilisées pour faire correspondre des ID utilisateur pour les données d'identification de sécurité Connect:Direct ou pour fractionner un fichier en plusieurs messages en créant un nouveau message chaque fois qu'une expression régulière est mise en correspondance. La syntaxe d'expression régulière utilisée par Managed File Transfer est la syntaxe prise en charge par l'API `java.util.regex`. Cette syntaxe d'expression régulière est similaire, mais pas identique, à la syntaxe d'expression régulière utilisée par le langage Perl.

Pour plus d'informations sur les expressions régulières Java, voir le Java tutoriel [Expressions régulières](#).

Exemples

Pour faire correspondre tous les modèles, utilisez l'expression régulière suivante:

```
.*
```

Pour faire correspondre tous les modèles commençant par la chaîne `fte`, utilisez l'expression régulière suivante:

```
fte.*
```

Pour faire correspondre tous les modèles commençant par la chaîne `accounts` suivie d'un chiffre unique et se terminant par `.txt`, utilisez l'expression régulière suivante:

```
accounts[0-9]\.txt
```

Variables de substitution à utiliser avec les processus Connect:Direct définis par l'utilisateur

Vous pouvez définir des valeurs pour remplacer des processus Connect:Direct définis par l'utilisateur à l'aide de variables symboliques intrinsèques spécifiques à Managed File Transfer.

Pour respecter la convention de dénomination Connect:Direct, toutes les variables symboliques intrinsèques utilisées par Managed File Transfer ont le format `%FTE` suivi de cinq caractères alphanumériques en majuscules. Pour plus d'informations sur les variables symboliques intrinsèques, voir la documentation du produit Connect:Direct.

Lors de la création d'un processus pour transférer des fichiers depuis un noeud Connect:Direct sur le système de pont Connect:Direct, vous devez utiliser la variable intrinsèque `%FTETFILE` comme valeur pour TO FILE dans le processus Connect:Direct. Lors de la création d'un processus pour transférer des fichiers sur un noeud Connect:Direct depuis le système de pont Connect:Direct, vous devez utiliser la variable intrinsèque `%FTEFFILE` comme valeur de FROM FILE dans le processus Connect:Direct. Ces variables contiennent les chemins d'accès aux fichiers temporaires que l'agent de pont Connect:Direct utilise pour les transferts vers et depuis le réseau Managed File Transfer.

Tableau 69. Variables symboliques intrinsèques utilisées par Managed File Transfer et Connect:Direct

Nom de la variable	Description
<code>%FTESAGNT</code>	Nom de l'agent source Managed File Transfer. Cette variable est définie uniquement pour les transferts d'un noeud Managed File Transfer Agent vers un noeud Connect:Direct.
<code>%FTEDAGNT</code>	Nom de l'agent de destination Managed File Transfer. Cette variable est définie uniquement pour les transferts d'un noeud Connect:Direct vers un Managed File Transfer Agent.
<code>%FTEPNODE</code>	Nom du noeud principal Connect:Direct. La valeur est toujours le nom du noeud Connect:Direct qui fait partie du pont Connect:Direct.
<code>%FTEPPLAT</code>	Plateforme sur laquelle s'exécute le noeud principal Connect:Direct. Les valeurs possibles pour cette variable sont UNIX et WINDOWS. Ces informations sont fournies par l'agent de pont Connect:Direct.
<code>%FTEPUSER</code>	Identificateur utilisateur du noeud principal Connect:Direct à utiliser dans le processus Connect:Direct. Ces informations sont extraites du fichier <code>ConnectDirectCredentials.xml</code> .
<code>%FTEPPASS</code>	Mot de passe à utiliser avec le nom d'utilisateur défini par la variable <code>%FTEPUSER</code> . Ces informations sont extraites du fichier <code>ConnectDirectCredentials.xml</code> .
<code>%FTESNODE</code>	Nom du noeud secondaire Connect:Direct. La valeur est toujours le nom du noeud Connect:Direct vers lequel ou depuis lequel le fichier est transféré.
<code>%FTESPLAT</code>	Plateforme sur laquelle s'exécute le noeud secondaire Connect:Direct. Les valeurs possibles pour cette variable sont UNIX, WINDOWS et ZOS. Ces informations sont extraites du fichier <code>ConnectDirectNodeProperties.xml</code> .
<code>%FTESUSER</code>	Identificateur utilisateur du noeud secondaire Connect:Direct à utiliser dans le processus Connect:Direct. Ces informations sont extraites du fichier <code>ConnectDirectCredentials.xml</code> .

Tableau 69. Variables symboliques intrinsèques utilisées par Managed File Transfer et Connect:Direct (suite)

Nom de la variable	Description
%FTESPASS	Mot de passe à utiliser avec le nom d'utilisateur défini par la variable %FTESUSER. Ces informations sont extraites du fichier ConnectDirectCredentials.xml .
%FTEFFILE	<p>Nom du fichier source. Cette variable est définie uniquement pour les processus Connect:Direct soumis dans une portée par fichier.</p> <p>Lors du transfert de fichiers d'un Managed File Transfer Agent vers un noeud Connect:Direct , la valeur correspond à l'emplacement qualifié complet du fichier sur le même système que le pont Connect:Direct .</p> <p>Lors du transfert de fichiers d'un noeud Connect:Direct vers un Managed File Transfer Agent, la valeur est le nom du fichier spécifié comme fichier source dans la demande de transfert Managed File Transfer .</p>
%FTEFDISP	<p>Disposition du fichier source lorsque le processus est terminé. La valeur de cette variable dépend de la plateforme et est équivalente aux valeurs de la demande de transfert MFT . Cette variable est définie uniquement pour les processus Connect:Direct soumis dans une portée par fichier.</p> <p>Lors du transfert de fichiers d'un Managed File Transfer Agent vers un noeud Connect:Direct , l'action de suppression ou de non suppression du fichier source est effectuée par l'agent de pont Managed File Transfer .</p> <p>Lors du transfert de fichiers d'un noeud Connect:Direct vers un Managed File Transfer Agent, l'action de suppression ou de non suppression du fichier source doit être effectuée par le processus Connect:Direct .</p>
%FTEFCP	<p>Page de codes à utiliser pour le fichier source. Cette variable est définie uniquement pour les processus Connect:Direct soumis dans une portée par fichier.</p> <p>Lors du transfert de fichiers d'un Managed File Transfer Agent vers un noeud Connect:Direct , cette valeur est UTF-8 ou, si le transfert est un transfert binaire, elle n'est pas définie.</p> <p>Lors du transfert de fichiers d'un noeud Connect:Direct vers un Managed File Transfer Agent, cette valeur est spécifiée par Connect:Direct ou, si le transfert est un transfert binaire, elle n'est pas définie.</p>
%FTEFSYSO	<p>Connect:Direct SYSOPTS pour la source du transfert. Si le noeud Connect:Direct distant se trouve sous Linux, UNIXou Windows, cette valeur contient des informations sur la page de codes et le type de données de la source du transfert.  Si le noeud distant est sous z/OS, cette valeur contient des informations supplémentaires.</p>
%FTEFNODE	Identifie le noeud Connect:Direct sur lequel réside le fichier source. Elle sera définie sur la valeur PNODE ou SNODE.

Tableau 69. Variables symboliques intrinsèques utilisées par Managed File Transfer et Connect:Direct (suite)

Nom de la variable	Description
%FTETFILE	<p>Nom du fichier de destination. Cette variable est définie uniquement pour les processus Connect:Direct soumis dans une portée par fichier.</p> <p>Lors du transfert de fichiers d'un Managed File Transfer Agent vers un noeud Connect:Direct , la valeur est le nom du fichier spécifié comme fichier de destination dans la demande de transfert Managed File Transfer .</p> <p>Lors du transfert de fichiers d'un noeud Connect:Direct vers un Managed File Transfer Agent, la valeur correspond au nom qualifié complet de l'emplacement dans lequel écrire le fichier sur le même système que le pont Connect:Direct .</p>
%FTETDISP	<p>Disposition du fichier de destination. La valeur de cette variable dépend de la plateforme et est équivalente aux valeurs de la demande de transfert Connect:Direct . Cette variable est définie uniquement pour les processus Connect:Direct soumis dans une portée par fichier.</p> <p>Lors du transfert de fichiers d'un Managed File Transfer Agent vers un noeud Connect:Direct , l'action de création d'un fichier ou de remplacement d'un fichier existant doit être effectuée par le processus Connect:Direct .</p> <p>Lors du transfert de fichiers d'un noeud Connect:Direct vers un Managed File Transfer Agent, l'action de création d'un fichier ou de remplacement d'un fichier existant est effectuée par l'agent de pont Managed File Transfer .</p>
%FTETCP	<p>Page de codes à utiliser pour le fichier de destination. Cette variable est définie uniquement pour les processus Connect:Direct soumis dans une portée par fichier.</p> <p>Lors du transfert de fichiers d'un Managed File Transfer Agent vers un noeud Connect:Direct , cette valeur est spécifiée par Connect:Direct ou, si le transfert est un transfert binaire, elle n'est pas définie.</p> <p>Lors du transfert de fichiers d'un noeud Connect:Direct vers un Managed File Transfer Agent, cette valeur est UTF-8 ou, si le transfert est un transfert binaire, la valeur n'est pas définie.</p>
%FTETSYSO	<p>Connect:Direct SYSOPTS pour la destination du transfert. Si le noeud Connect:Direct distant est sous UNIX, Connect:Direct ou Windows, cette valeur contient des informations sur la page de codes et le type de données de la destination du transfert.  Si le noeud distant est sous Windows, cette valeur contient des informations supplémentaires.</p>
%FTETNODE	<p>Identifie le noeud Connect:Direct sur lequel le fichier de destination doit résider. Elle sera définie sur la valeur PNODE ou SNODE.</p>
%FTEDTYPE	<p>Type ou mode de données du transfert. Les valeurs possibles pour cette variable sont text ou binary. Cette variable est définie uniquement pour les processus Connect:Direct soumis dans une portée par fichier.</p>
%FTETRID	<p>ID de transfert hexadécimal de 48 caractères provenant du transfert Managed File Transfer .</p>
%FTEJOB	<p>Nom de travail de la demande de transfert Managed File Transfer . La valeur de cette variable est tronquée à 256 caractères et peut être utilisée dans les données comptables du processus.</p>

Tableau 69. Variables symboliques intrinsèques utilisées par Managed File Transfer et Connect:Direct (suite)

Nom de la variable	Description
%FTEPNAME	Nom de processus Connect:Direct généré par l'agent de pont Managed File Transfer . La valeur de cette variable est de 8 caractères alphanumériques. La valeur commence toujours par un caractère alphabétique.
%FTEMETA (clé)	Métadonnées de la demande de transfert Managed File Transfer . La valeur de <i>key</i> est la clé des métadonnées. La valeur de <i>key</i> n'est pas sensible à la casse. Une clé de ABC est traitée comme une clé de abc. Si ABC et abc sont définis en tant que clés de métadonnées, la valeur des secondes métadonnées définies remplace la valeur des premières métadonnées définies.

 Le tableau suivant contient des informations sur les variables symboliques intrinsèques supplémentaires qui sont utilisées lorsque le noeud Connect:Direct distant dans le transfert se trouve sur une plateforme z/OS .

Tableau 70.

Nom de la variable	Description
%FTEFDCB	Valeur du paramètre DCB à la source du transfert.
%FTEFSPACE	Valeur du paramètre SPACE à la source du transfert.
%FTEFLBEL	Valeur du paramètre LABEL à la source du transfert.
%FTEFUNIT	Valeur du paramètre UNIT à la source du transfert.
%FTEFVOL	Valeur du paramètre VOL à la source du transfert.
%FTEFDAACL	Valeur du paramètre DATACLAS à la source du transfert.
%FTETDCB	Valeur du paramètre DCB à la destination du transfert.
%FTETSPACE	Valeur du paramètre SPACE à la destination du transfert.
%FTETLBEL	Valeur du paramètre LABEL à la destination du transfert.
%FTETUNIT	Valeur du paramètre UNIT à la destination du transfert.
%FTETVOL	Valeur du paramètre VOL à la destination du transfert.
%FTETDAACL	Valeur du paramètre DATACLAS à la destination du transfert.
%FTETDSTY	Valeur du paramètre DSNTYPE à la destination du transfert.
%FTETLIKE	Valeur du paramètre LIKE à la destination du transfert.
%FTETMGCL	Valeur du paramètre MGMTCLAS à la destination du transfert.

Tableau 70. (suite)	
Nom de la variable	Description
%FTETSTCL	Valeur du paramètre STORCLAS à la destination du transfert.

Exemple: un fichier de processus Connect:Direct qui appelle des commandes MFT

Exemple de fichier de processus Connect:Direct qui appelle la commande Managed File Transfer **ftetag** et la commande **ftecxfer**.

Dans cet exemple, les actions suivantes se produisent:

1. Une instruction Connect:Direct COPY transfère le fichier de C:\test\from\sent.txt sur le système où s'exécute le noeud secondaire vers C:\test\tmp\midpoint.txt sur le système où s'exécute le noeud principal.
2. Le processus Connect:Direct appelle la commande **ftetag** pour créer des informations d'audit dans MFT.
3. Le processus Connect:Direct appelle la commande **ftecxfer**.
4. La commande **ftecxfer** transfère le fichier de C:\test\tmp\midpoint.txt sur le système où s'exécute le noeud principal et où s'exécute l'agent CD_BRIDGE vers /test/to/arrived.txt sur le système où se trouve l'agent LINUX_AGENT.

```

/*BEGIN_REQUESTER_COMMENTS
 $PNODE$="cd_win01" $PNODE_OS$="Windows"
 $SNODE$="CD_WIN01" $SNODE_OS$="Windows"
 $OPTIONS$="WDOS"
END_REQUESTER_COMMENTS*/

TESTPRO PROCESS
  SNODE=CD_WIN01

COPY
  FROM (
    FILE=C:\test\from\sent.txt
    SNODE
  )
  TO (
    FILE=C:\test\tmp\midpoint.txt
    PNODE
    DISP=RPL
  )
  COMPRESS Extended

RUN TASK PNODE
  SYSOPTS="pgm(C:\wmqfte\bin\ftetag) args(C:\test\tmp\midpoint.txt)"

RUN TASK PNODE
  SYSOPTS="pgm(C:\wmqfte\bin\ftecxfer) args(-qmgrname QM_CDBA -connname fish.example.com(1441)
  -channelname SYSTEM.DEF.SVRCONN
  -sa CD_BRIDGE -da LINUX_AGENT -sm QM_CDBA -dm QM_LINUX -de overwrite -df /test/to/arrived.txt
  C:\test\tmp\midpoint.txt"

PEND

```

Concepts associés

«Utilisation de processus Connect:Direct pour soumettre des demandes de transfert Managed File Transfer», à la page 249

Vous pouvez soumettre une demande de transfert à l'agent de pont Connect:Direct à partir d'un processus Connect:Direct. Managed File Transfer fournit des commandes qui peuvent être appelées à partir d'une instruction **RUN TASK** dans un processus Connect:Direct.

Tâches associées

«Création et soumission d'un processus Connect:Direct qui appelle Managed File Transfer à l'aide de Connect:Direct Requester», à la page 250

Le demandeur Connect:Direct est une interface graphique que vous pouvez utiliser pour créer et soumettre un processus Connect:Direct qui appelle Managed File Transfer.

Restrictions de l'agent de pont Connect:Direct

L'agent de pont Connect:Direct est configuré pour transférer des fichiers vers et depuis des noeuds Connect:Direct . Il existe des fonctions que l'agent de pont Connect:Direct n'est pas capable d'exécuter.

- L'agent de pont Connect:Direct ne peut pas lire les messages d'une file d'attente ou écrire des messages dans une file d'attente. Il ne peut pas agir en tant qu'agent de destination dans un transfert de fichier à message ou en tant qu'agent source dans un transfert de message à fichier.
- Vous ne pouvez pas définir de moniteur de ressources sur l'agent de pont Connect:Direct .
- Vous ne pouvez pas avoir un agent de pont Connect:Direct comme source et destination d'un transfert. Vous ne pouvez pas effectuer de transfert depuis le noeud Connect:Direct vers le noeud Connect:Direct via le pont Connect:Direct .
- L'agent de pont Connect:Direct ne prend pas en charge les exits utilisateur appelés avant ou après le transfert. L'agent de pont Connect:Direct prend en charge un exit de mappage de données d'identification. Pour plus d'informations, voir la section [«Mappage des données d'identification pour Connect:Direct à l'aide de classes d'exit»](#), à la page 151.
- Vous ne pouvez pas définir des appels de programme presrc ou postsrc pour un transfert dont l'agent de pont Connect:Direct est l'agent source. Pour plus d'informations, voir la section [«Eléments imbriqués d'appel de programme»](#), à la page 894.
- Vous ne pouvez pas définir des appels de programme predst ou postdst pour un transfert dont l'agent de pont Connect:Direct est l'agent de destination. Pour plus d'informations, voir la section [«Eléments imbriqués d'appel de programme»](#), à la page 894.
- Vous ne pouvez pas spécifier de caractère générique dans la spécification source si l'agent source est l'agent de pont Connect:Direct .
- Si vous spécifiez une disposition de source (**-sd**) de `delete` lors du transfert d'un fichier  ou d'un ensemble de données à partir d'un noeud Connect:Direct , le comportement est différent du comportement de disposition de source habituel. L'un des cas suivants se produit:
 - Si Connect:Direct utilise un processus généré par Managed File Transfer pour déplacer le fichier ou l'ensemble de données depuis la source, l'indication de l'option `delete` entraîne l'échec du transfert. Pour spécifier que le fichier source doit être supprimé, vous devez soumettre un processus Connect:Direct défini par l'utilisateur. Pour plus d'informations, voir la section [«Soumission d'un processus Connect:Direct défini par l'utilisateur à partir d'une demande de transfert de fichier»](#), à la page 245.
 - Si Connect:Direct fait appel à un processus défini par l'utilisateur pour déplacer le fichier ou l'ensemble de données à partir de la source, ce paramètre est transmis au processus via la variable symbolique intrinsèque **%FTEFDISP**. Le processus défini par l'utilisateur détermine si la source est supprimé. Le résultat du transfert dépend du résultat renvoyé par le processus défini par l'utilisateur.

Concepts associés

[«Le pont Connect:Direct»](#), à la page 236

Vous pouvez transférer des fichiers depuis et vers un réseau IBM Sterling Connect:Direct existant. Utilisez le pont Connect:Direct, qui est un composant de Managed File Transfer, pour transférer des fichiers entre MFT et IBM Sterling Connect:Direct.

Prise en charge du serveur FTPS par le pont de protocole

Le pont de protocole prend en charge un sous-ensemble du protocole FTPS tel que défini par RFC-2228, RFC-4217 et Internet-Draft intitulé *Secure FTP over SSL*.

Pour obtenir la liste des valeurs de suite de chiffrement valides pour les connexions entre les agents de pont de protocole et les serveurs FTPS, voir [Cipher suites](#) dans la documentation du produit IBM SDK and Runtime Environment Java Technology Edition 7.

Les fonctions suivantes du protocole FTPS sont prises en charge:

- Modes de fonctionnement implicites et explicites.
- Validation du certificat serveur.
- Authentification mutuelle facultative à l'aide de vérifications de certificat client.
- Utilisation facultative d'un canal de contrôle d'effacement après la sélection de l'authentification initiale et du niveau de protection du canal de données.
- Les suites de chiffrement SHA-2 et la conformité à la norme FIPS 140-2 sont prises en charge. Les versions suivantes de Java sont requises: IBM JREs 6.0 SR13 FP2, 7.0 SR4 FP2 ou version ultérieure.

Les fonctions suivantes du protocole FTPS et de l'environnement d'exécution ne sont pas prises en charge:

- Utilisation de la commande **ADAT** pour un échange de données de sécurité supplémentaire.
- Utilisation de FTPS pour le chiffrement de canal uniquement, c'est-à-dire lorsque le certificat des serveurs n'est pas validé.
- Sélectionnez les niveaux de protection Effacer, Sécurisé ou Confidentiel à l'aide de la commande **PROT**.
- Chiffrement pour chaque commande à l'aide des commandes **MIC**, **CONF** et **ENC**.
- Rétromigration vers le protocole FTP si le serveur ne prend pas en charge FTPS explicite. Utilisez la prise en charge FTP fournie par le pont de protocole pour utiliser ce type de serveur.
- Utilisation de la commande **FEAT** pour déterminer les capacités disponibles du serveur FTPS.
- Validation des certificats à l'aide de la correspondance de modèle par rapport à la zone de nom distinctif.
- Vérification de la révocation de certificat.
- Validation des certificats auprès de l'autorité de certification de confiance émettrice.
- Sélection explicite des suites de chiffrement disponibles pour la phase de négociation SSL de l'établissement d'une session.
-  Utilisation d'extensions spécifiques à z/OS  ou IBM i qui intègrent la cryptographie au système d'exploitation. En particulier, l'utilisation du fichier de clés z/OS ou des systèmes de fichiers non hiérarchiques pour le stockage des informations de clé et de confiance, par exemple les fichiers. Le matériel de cryptographie et les moteurs de déchargement sont utilisés si ces fonctions sont gérées de manière transparente par la machine virtuelle Java et ne nécessitent pas de code d'application explicite.

Concepts associés

«Le pont de protocole», à la page 219

Le pont de protocole permet à votre réseau Managed File Transfer (MFT) d'accéder aux fichiers stockés sur un serveur de fichiers en dehors de votre réseau MFT, dans votre domaine local ou à un emplacement distant. Ce serveur de fichiers peut utiliser les protocoles de réseau FTP, FTPS ou SFTP. Chaque serveur de fichiers a besoin d'au moins un agent dédié. L'agent dédié est appelé agent de pont de protocole. Un agent de pont peut interagir avec plusieurs serveurs de fichiers.

Prise en charge du serveur SFTP par le pont de protocole

Le pont de protocole prend en charge le protocole SFTP tel que défini par l'IETF Internet Draft intitulé SSH File Transfer Protocol, version 6 draft 13.

Les agents de pont de protocole prennent en charge les chiffrements suivants lors de la connexion à un serveur de fichiers à l'aide du protocole SFTP:

- fclation-cbc
- 3des-cbc
- aes128-cbc
- aes192-cbc

- aes256-cbc
- aes128-ctr
- aes192-ctr
- aes256-ctr
- 3des-ctr
- arc quatre
- arcfour128
- arcfour256

Par défaut, la liste des chiffrements utilisés par les agents de pont de protocole est aes128-cbc,aes192-cbc,aes256-cbc. Pour plus d'informations sur la configuration d'un agent de pont de protocole pour qu'il utilise des chiffrements différents, voir [«Format du fichier de propriétés du pont de protocole»](#), à la page 592.

Méthodes d'authentification

Si vous avez fourni le code de l'agent de pont de protocole IBM MQ Managed File Transfer (MFT) avec une clé privée et un mot de passe de serveur, pour un utilisateur unique dans le fichier `ProtocolBridgeCredentials.xml`, l'agent de pont de protocole MFT configure par défaut la bibliothèque JSch pour qu'elle utilise les deux méthodes d'authentification, si le serveur de fichiers SFTP l'exige, lors de l'établissement d'une connexion.

Si une clé privée et un mot de passe de serveur sont configurés pour un seul utilisateur dans le fichier `ProtocolBridgeCredentials.xml`, mais que le serveur de fichiers SFTP ne requiert qu'une seule de ces méthodes d'authentification, l'agent de pont de protocole MFT configure la bibliothèque JSch pour utiliser l'authentification par clé publique / privée de préférence à l'authentification par mot de passe.

Si le serveur de fichiers SFTP rejette la tentative d'utilisation de l'authentification par clé publique / privée, l'agent de pont de protocole MFT, utilisant la bibliothèque JSch, tente l'authentification par nom d'utilisateur et mot de passe.

Si l'une ou l'autre de ces authentifications aboutit, une connexion est établie au serveur de fichiers SFTP.

Pour configurer à la fois une clé privée et une authentification par mot de passe pour le fichier `ProtocolBridgeCredentials.xml`, associé à l'agent de pont de protocole MFT, vous devez spécifier:

- L'attribut **serverPassword** (avec la valeur associée) dans l'élément qui mappe d'un nom d'utilisateur MFT à un nom d'utilisateur de serveur de protocole, et
- Élément de l'utilisateur MFT défini par l'élément parent.

Par exemple, la syntaxe peut être la suivante:

```
-----BEGIN RSA PRIVATE KEY-----
...
-----END RSA PRIVATE KEY-----
```

Méthode interactive à l'aide du clavier

L'agent de pont de protocole MFT utilise le JSch, bibliothèque tierce, pour se connecter aux serveurs de fichiers SFTP. Vous pouvez configurer la bibliothèque JSch de sorte qu'elle puisse tenter de s'authentifier auprès d'un serveur de fichiers SFTP à l'aide de la méthode *keyboard-interactive* lorsqu'aucune clé privée n'est spécifiée dans le fichier `ProtocolBridgeCredentials.xml`.

Notez que l'authentification à l'aide de la méthode *keyboard-interactive* fonctionne uniquement si le serveur de fichiers SFTP vous invite à entrer le mot de passe à l'aide de la chaîne `password:` (en majuscules, en minuscules ou en casse mixte). Dans le cas où vous utilisez la méthode d'authentification *clavier-interactif* et que le serveur de fichiers SFTP répond avec une chaîne différente de `password:`, la tentative de connexion échoue.

Lorsque le serveur de fichiers SFTP répond à la tentative de connexion initiale avec cette chaîne, l'agent de pont de protocole, à l'aide de la bibliothèque JSch, envoie le mot de passe configuré dans l'attribut **serverPassword** de l'élément `user` dans le fichier `ProtocolBridgeCredentials.xml`.

Information associée

[Le pont de protocole](#)

Prise en charge de FIPS dans MFT

Managed File Transfer prend en charge l'utilisation de modules de cryptographie compatibles FIPS dans les connexions client à partir des agents, des commandes et de l'explorateur IBM MQ vers les gestionnaires de files d'attente. Toutes les connexions SSL au gestionnaire de files d'attente utilisent uniquement le protocole TLS. La prise en charge est fournie pour les types de magasin de clés JKS et PKCS#12.

Indiquez si vous souhaitez activer la prise en charge FIPS pour un agent, un gestionnaire de files d'attente de coordination ou un gestionnaire de files d'attente de commandes, comme suit:

- Si vous souhaitez activer la norme FIPS pour un agent spécifique, définissez les propriétés `agentSsl` appropriées dans le fichier `agent.properties` de cet agent. Pour plus d'informations, voir la section [«Propriétés SSL pour MFT»](#), à la page 622.
- Si vous souhaitez activer FIPS pour un gestionnaire de file d'attente de coordination spécifique, définissez les propriétés `coordinationSsl` appropriées dans le fichier `coordination.properties` pour ce gestionnaire de file d'attente de coordination. Pour plus d'informations, voir la section [«Propriétés SSL pour MFT»](#), à la page 622.
- Si vous souhaitez activer FIPS pour un gestionnaire de files d'attente de commandes spécifique, définissez les propriétés `connectionSsl` appropriées dans le fichier `command.properties` pour ce gestionnaire de files d'attente de commandes. Pour plus d'informations, voir la section [«Propriétés SSL pour MFT»](#), à la page 622.

 FIPS n'est pas pris en charge sur Managed File Transfer for  IBM i.

FIPS n'est pas pris en charge sur les connexions vers ou depuis un pont de protocole ou un pont Connect:Direct.

Pour plus d'informations sur IBM MQ et FIPS et les étapes de configuration requises, voir [Federal Information Processing Standards \(FIPS\)](#).

Si vous souhaitez utiliser FIPS, la CIPHERSuite doit être conforme à la norme FIPS ou la connexion doit échouer. Pour plus d'informations sur les CIPHERSpecs pris en charge par IBM MQ, voir [SSL/TLS CIPHERSpecs et CIPHERSuites dans IBM MQ classes for Java](#) et [SSL/TLS CIPHERSpecs et CIPHERSuites dans IBM MQ classes for JMS](#).

Tables du consignateur de base de données MFT

Une fois que vous avez installé et configuré le consignateur, les tables de base de données suivantes sont créées:

AUTH_EVENT

Événement lié à la vérification des droits, généralement le rejet d'une demande en raison de privilèges insuffisants.

- **ID:** ID ligne.
- **ACTION:** Type d'action qui a eu lieu.
- **COMMAND_ID:** ID message IBM MQ du message d'origine qui a demandé l'événement. Dans le cas d'une demande de transfert, il s'agit également de l'ID de transfert.
- **TIME:** heure à laquelle l'événement s'est produit.
- **ORIGINATOR_MQ_USER:** ID utilisateur contenu dans le message IBM MQ, sur lequel la vérification des droits a été effectuée.

- **AUTHORITY:** droits requis pour l'action demandée.
- **ORIGINAL_XML_REQUEST:** Contenu du message de commande indiquant l'action qui a été refusée.
- **RESULTCODE:** Code numérique identifiant le résultat.
- **RESULT_TEXT:** Message expliquant le résultat de l'événement de droits d'accès.

CALL

Exécution à distance d'une commande du système d'exploitation, d'un script Ant  ou d'un travail JCL z/OS géré par Managed File Transfer. Les appels peuvent être imbriqués dans des transferts ou référencés par des lignes call_request.

Un CALL (c'est-à-dire une ligne de cette table) peut faire partie d'un transfert normal (auquel cas TRANSFER_CALLS est utilisé pour le lier à l'entrée pertinente dans TRANSFERS) ou il peut s'agir d'un appel géré autonome seul (disponible uniquement à partir de Ant ou en insérant directement des messages). Dans ce dernier cas, la table CALL_REQUEST est utilisée à la place de la table TRANSFERS ; un équivalent à TRANSFER_CALLS n'est pas nécessaire car il ne peut y avoir qu'un seul appel par demande d'appel.

- **ID:** ID ligne.
- **COMMAND:** commande qui a été exécutée. Cette zone ne contient aucun argument transmis à la commande ou au chemin dans lequel se trouve la commande.
- **TYPE:** Type de commande, par exemple Ant ou JCL.
- **RETRIES:** nombre de nouvelles tentatives demandées.
- **RETRY_WAIT:** Intervalle d'attente entre les nouvelles tentatives demandées à l'origine, en secondes.
- **SUCCESS_RC:** Code retour indiquant que la commande a abouti. Si un autre code est reçu, l'exécution est signalée comme ayant échoué.
- **EXECUTED_COMMAND:** nom complet de la commande qui a été exécutée, y compris le chemin d'accès.
- **CAPPED_RETRIES:** nombre de nouvelles tentatives disponibles ; ce nombre peut être inférieur à la valeur demandée si le nombre maximal de nouvelles tentatives de l'agent est inférieur au nombre de nouvelles tentatives demandées.
- **CAPPED_RETRY_WAIT:** intervalle entre les nouvelles tentatives utilisées ; ce nombre peut être inférieur à la valeur demandée si la limite configurée de l'agent est inférieure à l'attente de nouvelles tentatives demandée.
- **Résultat:** Indique si l'appel a abouti dans l'ensemble. S'il y a eu plusieurs tentatives, le résultat de chacune d'elles est enregistré séparément dans la table CALL_RESULT.

CALL_ARGUMENT

Argument ou paramètre fourni à une commande appelée.

- **ID:** ID ligne.
- **CALL_ID:** Appel auquel l'argument est associé.
- **KEY:** où l'argument est de type key-value-pair, clé ou nom.
- **TYPE:** Type de l'argument: certains sont des paramètres de position des commandes du système d'exploitation et d'autres sont des propriétés nommées utilisées avec Ant.
- **VALUE:** valeur de l'argument.

REQUÈRE_APPEL

Véhicule d'un appel de commande qui ne fait pas partie d'un transfert de fichier. Vous pouvez soumettre des messages ManagedCall à l'aide de Ant et de l'injection XML directe.

- **ID:** ID hexadécimal de la demande d'appel géré.

- **CALL_ID:** ID de la base de données de la ligne de la table CALL décrivant cet appel.
- **ACTION_TIME:** heure à laquelle l'action s'est produite.
- **AGENT:** agent sur lequel la commande est exécutée.
- **AGENT_QM:** gestionnaire de files d'attente utilisé par l'agent sur lequel la commande est exécutée.
- **ARCHITECTURE:** Architecture machine du système sur lequel l'agent s'exécute.
- **OS_NAME:** Nom du système d'exploitation sur lequel l'agent s'exécute.
- **OS_VERSION:** Version du système d'exploitation.
- **ORIGINATOR_HOST:** nom d'hôte de la machine à partir de laquelle la demande d'appel a été soumise.
- **ORIGINATOR_USER:** Nom de l'utilisateur qui a soumis la demande d'appel, comme indiqué dans le code XML de la demande.
- **ORIGINATOR_MQ_USER:** Nom de l'utilisateur qui a soumis la demande d'appel, tel qu'il figure dans le descripteur de message IBM MQ de la demande.
- **JOB_NAME:** nom de travail spécifié par l'utilisateur.
- **RESULTCODE:** Code de résultat global de l'appel.
- **RESULTTEXT:** Message de résultat global pour l'appel.

CALL_RESULT

Résultat détaillé de l'appel d'une commande. Un appel peut avoir plusieurs résultats si les nouvelles tentatives ont été activées.

- **ID:** ID ligne.
- **CALL_ID:** ID de la base de données de la ligne de la table CALL à laquelle ce résultat s'applique.
- **SEQUENCE:** La tentative à laquelle ce résultat s'applique, lorsqu'il y a eu plusieurs tentatives.
- **Résultat:** Résultat (par exemple, réussite ou échec) de la commande.
- **RETURN_CODE:** Code retour de la commande.
- **TIME:** heure à laquelle la commande s'est terminée.
- **STDOUT:** flux de sortie standard de la commande, s'il a été démarré.
- **STDERR:** Flux d'erreur standard de la commande, s'il a été démarré.
- **ERREUR:** Si la commande n'a pas pu être démarrée, un message d'erreur généré par Managed File Transfer expliquant le problème.

FILE_SPACE_ENTRY

Chaque ligne représente un fichier qui a été envoyé à l'espace fichier nommé.

- **ID:** ID de l'entrée d'espace fichier.
- **FILE_SPACE_NAME:** nom de l'espace fichier. Il s'agit du nom de l'utilisateur auquel appartient l'espace fichier.
- **TRANSFER_ITEM_ID:** ID de l'élément de transfert auquel cette ligne est associée.
- **ALIAS:** Nom d'alias de cette entrée d'espace fichier. En règle générale, ce nom d'alias correspond au nom du fichier source pour le transfert.
- **DELETED:** Heure à laquelle le fichier a été supprimé de l'espace fichier. Si le fichier n'a pas été supprimé, la valeur est null.

METADATA

Métadonnées associées à un transfert.

- **ID:** ID ligne.

- **TRANSFER_EVENT_ID:** Ligne transfer_event à laquelle ces métadonnées sont associées, si elle est associée à un transfert. Cette zone est null si les métadonnées sont associées à un appel géré autonome.
- **STANDALONE_CALL_ID:** Si les métadonnées sont associées à un appel géré autonome, ID de la demande d'appel géré concernée.
- **KEY:** nom de l'élément de métadonnées.
- **VALUE:** Valeur de l'élément de métadonnées.

MONITOR

Moniteurs de ressources qui déclenchent des opérations Managed File Transfer en fonction de conditions externes.

- **AGENT:** Agent sur lequel s'exécute le moniteur.
- **ID:** ID hexadécimal du moniteur.
- **NOM:** Nom du moniteur.
- **QMGR:** Gestionnaire de files d'attente de l'agent sur lequel s'exécute le moniteur.

MONITOR_ACTION

Chaque ligne représente une action (par exemple, la création et le déclenchement) qui se produit pour un moniteur

- **ID:** ID ligne.
- **ACTION:** Type d'action qui a eu lieu.
- **JOB_NAME:** nom du travail soumis, le cas échéant.
- **MONITOR:** Moniteur sur lequel cette action s'est produite. Peut être null si l'action a échoué car elle a été demandée pour un moniteur qui n'existe pas.
- **ORIGINAL_XML_REQUEST:** Si cette action était une action *create* ou *triggerSatisfied*, il s'agit de la demande XML qui est démarrée lorsque le moniteur est déclenché.
- **ORIGINATOR_MQ_USER:** ID utilisateur contenu dans le message IBM MQ qui a lancé l'action
- **ORIGINATOR_USER:** Nom d'utilisateur ayant soumis la demande d'exécution de l'action.
- **ORIGINATOR_HOST:** machine à partir de laquelle l'utilisateur a soumis la demande d'exécution de l'action.
- **TIME:** heure à laquelle l'action s'est produite.
- **UPDATED_XML_REQUEST:** Si l'action est *triggerSatisfied*, la demande XML qui a été démarrée. Cette demande peut être différente de la demande XML initialement effectuée en raison de la substitution de variables.

MONITOR_EXIT_RESULT

Résultat de l'exécution d'un exit de moniteur de ressources.

- **ID:** ID ligne.
- **ACTION_ID:** action de surveillance à laquelle le résultat est associé.
- **EXIT_NAME:** nom de l'exit qui a généré ce résultat.
- **RESULTCODE:** Valeur renvoyée par l'exit, à savoir annuler ou continuer.
- **RESULTTEXT:** Sortie texte de l'exit, si fournie.

MONITOR_METADATA

Éléments de métadonnées associés à un moniteur de ressources.

- **ID:** ID ligne.
- **ACTION_ID:** action de contrôle à laquelle les métadonnées sont associées.
- **KEY:** nom de l'élément de métadonnées.
- **PHASE:** indique si cet élément de métadonnées représente les données qui ont été soumises à l'origine ou la version mise à jour après la substitution de variable.
- **VALUE:** Valeur de l'élément de métadonnées.

SCHEDULE

Planification de transfert enregistrée auprès d'un agent.

- **AGENT:** nom de l'agent associé à cette planification.
- **CREATION_DATE:** moment auquel cette planification a été créée.
- **ID:** ID unique de la base de données (et non de l'agent) pour la planification.
- **ID_ON_AGENT:** ID utilisé par l'agent pour l'ID de base de données. Cet ID n'est pas unique entre les agents et peut même ne pas être unique dans un agent si l'état persistant de l'agent est réinitialisé.
- **LATEST_ACTION:** action la plus récente qui a modifié l'état de cette planification.

SCHEDULE_ACTION

Lorsqu'un événement qui modifie l'état de la planification se produit, une action est enregistrée.

- **ACTION_TYPE:** action qui s'est produite.
- **ID:** ID ligne
- **ORIGINATOR_HOST:** machine à partir de laquelle la demande à l'origine de la modification a été soumise.
- **ORIGINATOR_USER:** Utilisateur dont le nom de la demande à l'origine de la modification a été soumis.
- **SCHEDULE_ID:** planning auquel cette action s'applique.
- **SPEC_AFTER:** spécification de planification qui représente l'état de cette planification après l'exécution de l'action.
- **STATUS_CODE:** Code retour numérique décrivant le résultat de l'action
- **STATUS_TEXT:** Description textuelle du résultat de l'action. Généralement null si l'action a abouti.
- **TIME:** moment auquel l'action s'est produite

SCHEDULE_SPEC

Détails d'un transfert planifié individuel.

- **ID:** ID ligne.
- **DESTINATION_AGENT:** Agent vers lequel les fichiers sont transférés.
- **DESTINATION_QM:** gestionnaire de files d'attente utilisé par l'agent de destination.
- **REPEAT_COUNT:** Nombre de fois à répéter si la planification se répète et est liée par le nombre d'occurrences au lieu d'une heure de fin.
- **REPEAT_FREQUENCY:** Nombre d'intervalles de répétition entre les transferts planifiés.
- **REPEAT_INTERVAL:** Si le transfert se répète, intervalle à partir duquel se répéter (par exemple, minutes ou semaines).
- **SOURCE_AGENT:** Agent à partir duquel les fichiers sont transférés.
- **SOURCE_QM:** gestionnaire de files d'attente utilisé par l'agent source.
- **START_TIME:** heure à laquelle aura lieu le premier transfert dans la planification.
- **START_TIMEBASE:** Base de temps pour les heures associées au transfert. Par exemple, indique s'il faut opérer à partir du fuseau horaire de l'agent ou à partir du fuseau horaire de l'administrateur.

- **START_TIMEZONE:** Fuseau horaire auquel correspond la base de temps et qui sera utilisé lors de l'exécution du planning.

SCHEDULE_ITEM

Chaque fichier (ou modèle à mettre en correspondance au moment du transfert) est représenté par un élément `schedule_item`.

- **ID:** ID ligne.
- **CHECKSUM_METHOD:** Mode de calcul du total de contrôle du fichier
- **DESTINATION_EXISTS_ACTION:** action effectuée par l'agent de destination si le fichier existe déjà sur la destination.
- **DESTINATION_FILENAME:** Fichier ou répertoire dans lequel les fichiers sont transférés.
- **DESTINATION_QUEUE:** Nom de la file d'attente de destination pour un transfert de fichier à message.
- **Multi DESTINATION_TYPE:** Indique si la colonne `destination_filename` fait référence à un fichier ou à un répertoire.
- **z/OS DESTINATION_TYPE:** Indique si la colonne `destination_filename` fait référence à un fichier, à un répertoire ou à un ensemble de données.
- **FILE_MODE:** mode (par exemple, *texte* ou *binnaire*) dans lequel le fichier est transféré.
- **RECURSIVE:** Lorsque l'agent crée le transfert en fonction de la planification, indique si l'agent réutilise (Y) ou non (N) le répertoire source.
- **SCHEDULE_SPEC_ID:** spécification de planification à laquelle cet élément est associé.
- **SOURCE_DISPOSITION:** action à effectuer sur les fichiers source une fois le transfert terminé.
- **SOURCE_FILENAME:** Fichier source, nom de répertoire ou modèle.
- **SOURCE_QUEUE:** Nom de la file d'attente source pour un transfert de message vers un fichier

TRANSFER

Transfert unique d'un ou de plusieurs fichiers.

- **TRANSFER_ID:** ID hexadécimal du transfert.
- **JOB_NAME:** Nom de travail spécifié par l'utilisateur pour le transfert.
- **SCHEDULE_ID:** Si ce transfert est le résultat d'une planification, l'ID de ligne de la base de données de la planification concernée.
- **START_ID:** ID de ligne de `transfer_event` qui représente le début du transfert.
- **COMPLETE_ID:** ID de ligne de l'événement de transfert qui représente la fin du transfert.
- **RESULTCODE:** Code de résultat global pour le transfert. Les valeurs possibles pour cette colonne sont répertoriées dans la rubrique suivante: «Codes retour pour MFT», à la page 347. Ces codes s'appliquent à l'ensemble du transfert ; voir [TRANSFER_ITEM.RESULTCODE](#) pour le statut de chaque élément individuel.
- **RESULTTEXT:** Texte de résultat global pour le transfert, le cas échéant.
- **STATUS:** statut d'un transfert. Les valeurs possibles pour cette colonne sont les suivantes: démarré, réussite, réussite partielle, échec et annulé.
- **RELATED_TRANSFER_ID:** ID hexadécimal d'un transfert précédent associé à ce transfert. Par exemple, si le transfert est un téléchargement de fichier, cette zone fait référence au transfert qui a transféré le fichier.

TRANSFER_CALLS

Lie les appels de commande exécutables aux transferts

- **ID:** ID ligne.

- **POST_DESTINATION_CALL:** Appel effectué à la destination une fois le transfert terminé.
- **POST_SOURCE_CALL:** Appel effectué au niveau de l'agent source une fois le transfert terminé.
- **PRE_DESTINATION_CALL:** Appel effectué au niveau de l'agent de destination avant le début du transfert.
- **PRE_SOURCE_CALL:** Appel effectué au niveau de l'agent source avant le début du transfert.
- **TRANSFER_ID:** transfert auquel sont associés les appels de cette ligne.

TRANSFER_CD_NODE

Informations sur les noeuds Connect:Direct utilisés dans un transfert.

- **PNODE:** noeud principal dans le transfert.
- **SNODE:** noeud secondaire dans le transfert.
- **BRIDGE_IS_PNODE:** Caractère indiquant le noeud qui fait partie du pont Connect:Direct . Si cette valeur est Y, le noeud principal est le noeud de pont. Si cette valeur est N, le noeud secondaire est le noeud de pont.
- **ID:** ID de cette ligne.

TRANSFER_CORRELATOR

Chaque ligne contient une chaîne de corrélation et un nombre associé à un élément de transfert.

- **CORRELATION_BOOLEAN:** Valeur de corrélation booléenne. Représenté par un caractère unique Y pour true et N pour false.
- **CORRELATION_STRING:** Valeur de corrélation de chaîne.
- **CORRELATION_NUMBER:** Valeur de corrélation numérique.
- **ID:** ID de cette ligne.

TRANSFER_EVENT

Événement (début ou fin) lié à un transfert.

- **ID:** ID ligne.
- **ACTION_TIME:** heure à laquelle l'action de transfert a eu lieu.
- **SOURCE_AGENT:** Nom de l'agent à partir duquel les fichiers sont transférés.
- **SOURCE_AGENT_TYPE:** Type d'agent à partir duquel les fichiers sont transférés. Les valeurs suivantes sont possibles: 1 = STANDARD, 2 = BRIDGE, 3 = WEB_GATEWAY, 4 = EMBEDDED, 5 = CD_BRIDGE, 6 = SFG.

Remarque : Depuis la IBM MQ 9.0, Managed File Transfer ne prend pas en charge la passerelle Web ou les agents Web.

- **SOURCE_QM:** gestionnaire de files d'attente utilisé par l'agent source.
- **SOURCE_ARCHITECTURE:** Architecture de la machine du système hébergeant l'agent source.
- **SOURCE_OS_NAME:** système d'exploitation de la machine de l'agent source.
- **SOURCE_OS_VERSION:** Version du système d'exploitation de la machine de l'agent source.
- **SOURCE_BRIDGE_URL:** Si l'agent source est un agent de pont de protocole, l'URL de la source de données vers laquelle il forme un pont.
- **SOURCE_CD_NODE_ID:** Noeud Connect:Direct qui est la source du transfert.
- **DESTINATION_AGENT:** Nom de l'agent vers lequel les fichiers sont transférés.
- **DESTINATION_AGENT_TYPE:** Type d'agent vers lequel les fichiers sont transférés. Les valeurs suivantes sont possibles: 1 = STANDARD, 2 = BRIDGE, 3 = WEB_GATEWAY, 4 = EMBEDDED, 5 = CD_BRIDGE, 6 = SFG.

Remarque : Depuis la IBM MQ 9.0, Managed File Transfer ne prend pas en charge la passerelle Web ou les agents Web.

- **DESTINATION_QM:** gestionnaire de files d'attente utilisé par l'agent de destination.
- **DESTINATION_BRIDGE_URL:** Si l'agent de destination est un agent de pont, l'URL de la source de données vers laquelle il forme un pont.
- **DESTINATION_CD_NODE_ID:** noeud Connect:Direct qui est la destination du transfert.
- **ORIGINATOR_HOST:** nom d'hôte de la machine à partir de laquelle la demande de transfert a été soumise.
- **ORIGINATOR_USER:** Nom de l'utilisateur qui a soumis la demande de transfert, comme indiqué par la commande `fteCreateTransfer`.
- **ORIGINATOR_MQ_USER:** Nom de l'utilisateur qui a soumis la demande de transfert, tel qu'il figure dans le descripteur de message IBM MQ de la demande.
- **TRANSFERSET_TIME:** heure à laquelle l'ensemble de transferts a été créé.
- **TRANSFERSET_SIZE:** nombre d'éléments en cours de transfert.
- **TRIGGER_LOG:** pour les définitions de transfert impliquant un déclencheur, indique si les évaluations de déclencheur qui n'ont pas donné lieu à un transfert doivent être consignées.

TRANSFER_EXIT

Chaque ligne représente un exit de transfert qui a été exécuté dans le cadre d'un transfert de fichiers.

- **ID:** ID ligne.
- **EXIT_NAME:** nom de l'exit.
- **TRANSFER_ID:** ID du transfert terminé ou annulé auquel s'applique cet exit.
- **TYPE:** Type de l'exit. Il peut s'agir de l'une des valeurs suivantes: *SourceStart*, *SourceEnd*, *DestinationStart* ou *DestinationEnd*.
- **STATUS:** valeur renvoyée par l'exit. Il peut s'agir de *annuler* ou de *continuer*.
- **SUPPLÉMENT:** Message facultatif expliquant le statut de l'exit.

TRANSFER_ITEM

Chaque ligne représente un fichier envoyé dans le cadre du transfert.

- **DESTINATION_CHECKSUM_METHOD:** algorithme utilisé pour calculer une somme de contrôle du fichier de destination. Peut être null si aucun total de contrôle n'a été calculé car le transfert n'a pas abouti.
- **DESTINATION_CHECKSUM_VALUE:** Valeur de total de contrôle du fichier de destination. La valeur peut être null si checksumming a été désactivé.
- **DESTINATION_ENCODING:** codage de caractères utilisé dans le fichier de destination, si le fichier de destination est transféré sous forme de texte.
- **DESTINATION_EXISTS_ACTION:** action à effectuer si le fichier existe sur la destination.
- **DESTINATION_FILE_SIZE:** Taille du nom de fichier `z/OS` ou du nom de fichier à utiliser à la destination.
- **DESTINATION_FILENAME:** nom de fichier `z/OS` ou nom de fichier à utiliser à la destination.
- **DESTINATION_LINEEND:** format de fin de ligne utilisé dans le fichier de destination, si le fichier de destination est transféré sous forme de texte.
- **DESTINATION_MESSAGE_QUEUE_NAME:** File d'attente de destination des messages générés à partir du fichier source lors d'un transfert de fichier à message.
- **DESTINATION_MESSAGE_GROUP_ID:** Si plusieurs messages sont générés, ID de groupe utilisé pour les messages générés à partir du fichier source lors d'un transfert de fichier à message.

- **DESTINATION_MESSAGE_MESSAGE_ID:** Si un seul message est généré, ID du message généré à partir du fichier source lors du transfert d'un fichier vers un message.
- **DESTINATION_MESSAGE_COUNT:** nombre de messages dans lesquels le fichier source a été fractionné lors d'un transfert de fichier à message.
- **DESTINATION_MESSAGE_LENGTH:** Longueur du message généré à partir du fichier source lors du transfert d'un fichier vers un message, en octets. Cette valeur n'est définie que si vous spécifiez une longueur pour les messages de sortie, par exemple à l'aide de l'option `-qs` de la commande **fteCreateTransfer**. Si vous spécifiez `-qs 20K` et que la taille de votre fichier source est de 50 Ko, les trois messages résultants sont de 20 Ko, 20 Ko et 10 Ko. Dans ce cas, la valeur de **DESTINATION_MESSAGE_LENGTH** est définie sur 20480.
- **DESTINATION_CORRELATOR_ID:** ID des informations de corrélateur pour la destination.
- **FILE_MODE:** mode de transfert de fichier, par exemple *text* ou *binary*.
- **ID:** ID ligne
- **RESULTCODE:** Code numérique indiquant le résultat du transfert de cet élément. Les valeurs possibles pour cette colonne sont répertoriées dans la rubrique suivante: «Codes retour pour les fichiers d'un transfert», à la page 353. Ces codes s'appliquent aux articles individuels du transfert ; voir **TRANSFER.RESULTCODE** pour le résultat du transfert dans son ensemble.
- **RESULT_TEXT:** explication textuelle du résultat du transfert. Généralement null si le transfert a abouti.
- **SOURCE_CHECKSUM_METHOD:** algorithme utilisé pour calculer une somme de contrôle du fichier source.
- **SOURCE_CHECKSUM_VALUE:** Valeur de total de contrôle du fichier source. La valeur peut être null si checksumming a été désactivé.
- **SOURCE_DISPOSITION:** action à effectuer sur le fichier source lorsque le transfert est terminé.
- **SOURCE_ENCODING:** codage de caractères utilisé dans le fichier source, si le fichier source est transféré sous forme de texte.
- **SOURCE_FILE_SIZE:** Taille du nom de fichier **z/OS** ou du nom de fichier à utiliser à la source.
- **SOURCE_FILENAME:** nom du fichier source **z/OS** ou nom du fichier .
- **SOURCE_LINEEND:** format de fin de ligne utilisé dans le fichier source, si le fichier source est transféré sous forme de texte.
- **SOURCE_MESSAGE_QUEUE_NAME:** File d'attente source des messages inclus dans le fichier de destination pour un transfert de message vers un fichier.
- **SOURCE_MESSAGE_GROUP_ID:** ID de groupe des messages inclus dans le fichier de destination pour un transfert de message à fichier.
- **SOURCE_MESSAGE_COUNT:** Nombre de messages inclus dans le fichier de destination pour un transfert de message à fichier.
- **SOURCE_CORRELATOR_ID:** ID des informations de corrélateur pour la source.
- **TRANSFER_ID:** transfert dont fait partie cet élément.
- **TRUNCATE_RECORDS:** indique si les enregistrements de fichier trop longs doivent être tronqués ou encapsulés.

TRANSFER_STATS

Ensemble de statistiques générées à la fin d'un transfert.

- **ID:** ID ligne.
- **TRANSFER_ID:** transfert auquel les statistiques font référence.
- **START_TIME:** heure à laquelle le transfert a commencé. Dans un système occupé ou disposant d'une connectivité intermittente, cette heure peut être postérieure à l'heure indiquée dans le message Démarré, car cette heure représente le point de début du traitement initial plutôt que le point de début du transfert des données.

- **RETRY_COUNT:** Nombre de fois où le transfert a dû être retenté en raison de problèmes de charge ou de disponibilité.
- **FILE_FAILURES:** nombre de fichiers dont le transfert a échoué.
- **FILE_WARNINGS:** Nombre de fichiers pour lesquels des avertissements ont été signalés lors de leur transfert.

TRIGGER_CONDITION

Une condition dans un transfert conditionnel Managed File Transfer de base. Par exemple, " file example.file exists".

- **ID:** ID ligne.
- **TRANSFER_EVENT_ID:** Événement de transfert auquel le déclencheur est associé.
- **CONDITION_TYPE:** Type de vérification utilisé dans le déclencheur. Par exemple, l'existence d'un fichier ou la taille d'un fichier.
- **COMPARAISON:** comparaison spécifique à effectuer. Par exemple, "supérieur ou égal à".
- **Valeur :** Valeur à laquelle comparer.
- **FILENAME:** Nom du fichier à examiner.

Concepts associés

[«Configuration d'un consignateur MFT», à la page 107](#)

Référence associée

[«fteStartLogger», à la page 542](#)

La commande **fteStartLogger** démarre une application de journalisation Managed File Transfer .

[«fteModifyLogger \(exécutez un consignateur MFT en tant que service Windows \)», à la page 504](#)

La commande **fteModifyLogger** permet de modifier un consignateur Managed File Transfer afin qu'il puisse être exécuté en tant que service Windows . Vous pouvez utiliser cette commande uniquement sur les plateformes Windows , vous devez être exécuté par un utilisateur qui est un administrateur IBM MQ et un membre du groupe mqm, et vous devez d'abord arrêter le consignateur à l'aide de la commande **fteStopLogger** .

[«fteStopLogger», à la page 546](#)

La commande **fteStopLogger** arrête un consignateur Managed File Transfer .

Droits d'accès au consignateur MFT

L'utilisateur du système d'exploitation qui exécute le consignateur requiert certains droits IBM MQ sur les files d'attente du consignateur et sur SYSTEM.FTE .

L'utilisateur du système d'exploitation qui exécute le consignateur requiert les droits IBM MQ suivants:

- CONNECT et INQUIRE sur le gestionnaire de files d'attente de coordination.
- Droit SUBSCRIBE sur le système SYSTEM.FTE .
- Droit PUT sur le système SYSTEM.FTE.LOG.RJCT.*nom_journal* file d'attente.
- Droit GET sur SYSTEM.FTE.LOG.CMD.*nom_journal* file d'attente.

Référence associée

[«Restriction des droits de groupe pour les ressources spécifiques à MFT», à la page 362](#)

Au lieu d'accorder des droits à des utilisateurs individuels pour tous les différents objets pouvant être impliqués, configurez deux groupes de sécurité pour l'administration du contrôle d'accès Managed File Transfer : FTEUSER et FTEAGENT. Il incombe à l'administrateur IBM MQ de créer et de remplir ces groupes. L'administrateur peut choisir d'étendre ou de modifier la configuration proposée décrite ici.

[«Restriction des droits utilisateur sur les actions de l'agent MFT», à la page 368](#)

En plus d'utiliser des groupes pour gérer l'accès aux ressources, vous pouvez activer un niveau supplémentaire de sécurité afin de restreindre les actions d'agent Managed File Transfer qu'un utilisateur

peut effectuer. Accordez des autorisations à un utilisateur dans une file d'attente d'autorisations d'agent pour permettre à l'utilisateur d'effectuer des actions d'agent spécifiques.

Droits d'accès aux fichiers de destination

Les droits d'accès aux fichiers de destination écrits par les agents de destination Managed File Transfer sont déterminés par la plateforme sur laquelle l'agent s'exécute.

Agents de destination sur les plateformes z/OS, UNIXet Linux



Vous devez modifier la valeur de **umask** sur votre système.

Par exemple, supposons que la valeur **umask** par défaut de votre ID utilisateur sur votre système z/OS est *0022*.

Lorsqu'un agent MFT s'exécute en tant que cet utilisateur et qu'il écrit un fichier de destination, le fichier dispose des droits suivants:

```
-IW-I--I--
```

Si vous remplacez la valeur de **umask** par, par exemple, *0006*, en exécutant la commande

```
umask 0006
```

et l'agent a redémarré, puis tous les fichiers de destination écrits par l'agent ont les droits suivants:

```
-IW-IW----
```

Notez que vous devez redémarrer l'agent après avoir exécuté la commande `umask` afin que l'agent puisse récupérer la nouvelle valeur.

Bien que z/OS soit utilisé comme exemple ici, les mêmes informations s'appliquent aux plateformes UNIXet Linux .

Agents de destination sous Windows



Par défaut, les droits d'accès sont hérités d'un dossier racine vers les fichiers et sous-dossiers en dessous, bien que cet héritage puisse être désactivé.

Votre administrateur Windows ou votre administrateur de domaine doit vérifier et gérer les droits et les modifier si nécessaire. Ils peuvent utiliser la commande `icacls` pour afficher, ajouter, mettre à jour et supprimer des droits.

Information associée

[Restriction des droits de groupe pour les ressources spécifiques à MFT](#)

[Restriction des droits utilisateur sur les actions de l'agent MFT](#)

Propriétés de message MQ définies par MFT sur les messages écrits dans les files d'attente de destination

Lors du transfert d'un fichier vers un message, Managed File Transfer peut définir des propriétés de message IBM MQ sur le premier message écrit dans la file d'attente de destination. Des propriétés de message IBM MQ supplémentaires sont définies lorsqu'un transfert de fichier à message a échoué.

Les propriétés de message IBM MQ permettent à une application de sélectionner des messages à traiter ou d'extraire des informations sur un message sans accéder aux en-têtes MQ Message Descriptor (MQMD) ou MQRFH2 . Voir [Propriétés de message](#).

Cette rubrique décrit le paramètre utilisé dans les commandes **fteCreateTransfer** et **fteCreateTemplate** pour indiquer que les propriétés de message doivent être ajoutées au premier message écrit dans la file d'attente de destination. Vous pouvez également spécifier que les propriétés de message doivent être ajoutées au premier message écrit dans la file d'attente de destination à l'aide de la valeur *dstmsgprop* du paramètre **fte:filespec**.

Propriétés standard

Vous pouvez utiliser le paramètre **-qmp** dans la commande **fteCreateTransfer** ou la commande **fteCreateTemplate** pour indiquer si les propriétés de message IBM MQ sont définies sur le premier message écrit dans la file d'attente de destination par le transfert. Pour un exemple d'utilisation de ce paramètre, voir la rubrique [«Exemple: Définition des propriétés de message IBM MQ sur un transfert de fichier à message»](#), à la page 208

Les propriétés de message IBM MQ contiennent des métadonnées de transfert. Les noms de propriété de message sont précédés du préfixe **usr.WMQFTE**. Le préfixe **usr.** rend ces propriétés de message disponibles pour les applications JMS.

usr.WMQFTETransferId

ID de transfert hexadécimal unique.

usr.WMQFTETransferMode

Type de transfert de fichier: mode binaire ou mode texte.

usr.WMQFTESourceAgent

Nom de l'agent source.

usr.WMQFTEDestinationAgent

Nom de l'agent de destination.

usr.WMQFTEFileName

Nom du fichier source.

usr.WMQFTEFileSize

Taille du fichier source en octets.

usr.WMQFTEFileLastModified

Heure de la dernière modification du fichier source. Cette valeur est exprimée en millisecondes, mesurée à partir de 00:00:00 UTC, 1er janvier 1970.

usr.WMQFTEFileIndex

Index du fichier en cours dans la liste des fichiers en cours de transfert. Le premier fichier de la liste a l'index 0.

usr.WMQFTEMqmdUser

ID utilisateur MQMD de l'utilisateur qui a soumis la demande de transfert.

Propriétés d'échec

Lorsqu'un transfert de fichier à message échoue après que l'agent de destination a écrit au moins un message dans la file d'attente de destination, Managed File Transfer écrit un message vide dans la file d'attente de destination. Si le paramètre **-qmp** est défini sur true, deux propriétés de message IBM MQ sont définies pour ce message vide. Pour obtenir un exemple d'échec de transfert de fichier à message, voir [«Echec d'un transfert de fichier à message»](#), à la page 211.

Lorsqu'un transfert de fichier à message échoue complètement, Managed File Transfer écrit un message vide dans la file d'attente de destination. Si le paramètre **-qmp** est défini sur true et que la longueur des données de message est supérieure à la valeur `maxInputOutputMessageLength`, le message d'erreur suivant s'affiche sur la ligne de commande.

```
Name WMQFTEResultCode
Value 40
Name WMQFTESupplement
Value BFGTR0072E: The transfer failed to complete due to the exception BFGI00205E:The message
```

```
data length 1290843 being written
to the output queue "M2F@q2" is greater than the maximum allowed 1048576.
```

Les propriétés de message IBM MQ contiennent des informations sur l'échec. Comme pour les propriétés de message standard, les noms de propriété de message sont préfixés avec **usr.WMQFTE** et sont disponibles pour les applications JMS.

usr.WMQFTEReturnCode

Code retour du transfert. Pour obtenir la liste des valeurs possibles pour ce code retour, voir la rubrique [«Codes retour pour MFT»](#), à la page 347.

usr.WMQFTESupplement

Message supplémentaire décrivant plus en détail la raison de l'échec du transfert.

Propriétés définies par l'utilisateur

Les métadonnées spécifiées à l'aide du paramètre **-md** avec la commande **fteCreateTransfer** peuvent être définies en tant que propriétés de message IBM MQ . Si le paramètre **-qmp** est défini sur true, toutes les métadonnées spécifiées par l'utilisateur sont ajoutées à l'en-tête du premier message.

Le nom des métadonnées est préfixé par **usr.** . Par exemple, si les métadonnées sont department=accounts, l'en-tête de message IBM MQ est défini sur **usr.department=accounts**.

Vous ne pouvez pas utiliser de métadonnées pour spécifier des en-têtes commençant par **usr.WMQFTE** ou **usr.com.ibm.wmqfte**. Si vous spécifiez des métadonnées dont le nom commence par **WMQFTE** ou **com.ibm.wmqfte**, ces métadonnées ne sont pas utilisées dans les propriétés de message et sont ignorées.

Concepts associés

[«Transfert de données de fichiers vers des messages»](#), à la page 203

Vous pouvez utiliser la fonction de fichier à message de Managed File Transfer pour transférer des données d'un fichier vers un ou plusieurs messages d'une file d'attente IBM MQ .

Tâches associées

[«Exemple: Définition des propriétés de message IBM MQ sur un transfert de fichier à message»](#), à la page 208

Vous pouvez utiliser le paramètre **-qmp** dans la commande **fteCreateTransfer** pour indiquer si les propriétés de message IBM MQ sont définies sur le premier message écrit dans la file d'attente de destination par le transfert. Les propriétés de message IBM MQ permettent à une application de sélectionner des messages à traiter ou d'extraire des informations sur un message sans accéder aux en-têtes IBM MQ Message Descriptor (MQMD) ou MQRFH2 .

Référence associée

[«Propriétés de message IBM MQ lues par MFT à partir des messages des files d'attente source»](#), à la page 701

L'agent qui lit les messages d'une file d'attente source dans un message pour le transfert de fichiers lit les propriétés de message IBM MQ dans le message. La valeur de ces propriétés peut être utilisée pour déterminer le comportement d'un transfert.

[«Codes retour pour MFT»](#), à la page 347

Les commandes Managed File Transfer , les tâches Ant et les messages de journal fournissent des codes retour indiquant si les fonctions ont abouti.

[«Echec d'un transfert de fichier à message»](#), à la page 211

Si un transfert de fichier à message échoue après que l'agent a commencé à écrire des données de fichier dans la file d'attente de destination, l'agent écrit un message dans la file d'attente pour indiquer à une application consommant les messages qu'un incident s'est produit.

[«fteCreateTransfer: démarrer un nouveau transfert de fichier»](#), à la page 447

La commande **fteCreateTransfer** crée et démarre un nouveau transfert de fichier à partir de la ligne de commande. Cette commande peut démarrer un transfert de fichier immédiatement, planifier un transfert de fichier à une heure et une date ultérieures, répéter un transfert planifié une ou plusieurs fois et déclencher un transfert de fichiers sous certaines conditions.

«fte: élément imbriqué Ant filespec», à la page 887

Le paramètre **fte:filespec** est utilisé en tant qu'élément imbriqué dans d'autres tâches. Utilisez

fte:filespec pour décrire un mappage entre un ou plusieurs fichiers source, répertoires  ou ensembles de données et une destination. Généralement, cet élément est utilisé lors de l'expression d'un ensemble de fichiers ou de répertoires  ou d'ensembles de données à déplacer ou à copier.

Propriétés de message IBM MQ lues par MFT à partir des messages des files d'attente source

L'agent qui lit les messages d'une file d'attente source dans un message pour le transfert de fichiers lit les propriétés de message IBM MQ dans le message. La valeur de ces propriétés peut être utilisée pour déterminer le comportement d'un transfert.

En-têtes utilisés pour annuler les transferts de messages vers des fichiers

Définissez les propriétés de message IBM MQ suivantes sur le dernier message d'un groupe pour annuler le message dans le transfert de fichier de ce groupe:

usr.UserReturnCode

Obligatoire. Code retour du transfert. Définissez cet en-tête comme une valeur non nulle pour indiquer que le transfert doit être annulé.

usr.UserSupplement

Facultatif. Texte décrivant la raison pour laquelle le transfert a été annulé.

Si l'agent source d'un transfert de message à fichier lit un message de la file d'attente source dont la propriété de message **usr.UserReturnCode** est définie sur une valeur différente de zéro, il arrête de lire les messages de la file d'attente et signale que le transfert a échoué dans le journal de transfert XML. Le fichier XML du journal de transfert contient le code retour et le texte supplémentaire définis dans les en-têtes de message. Si l'agent de destination a déjà écrit des données dans un fichier temporaire, ce fichier est supprimé de la destination.

En-têtes utilisés par la substitution de variable

La valeur d'une propriété de message IBM MQ dans le premier message à lire à partir de la file d'attente surveillée peut être remplacée dans la définition XML de la tâche. Les propriétés de message définies par l'utilisateur sont préfixées avec `usr.`, mais n'incluent pas ce préfixe dans le nom de la variable. Les noms de variable doivent être précédés d'un signe dollar (\$) et placés entre accolades ({ }). Par exemple, `${destFileName}` est remplacé par la valeur de la propriété de message `usr.destFileName` du premier message à lire dans la file d'attente source.

Par exemple, l'utilisateur ou le programme qui place des messages dans une file d'attente surveillée peut définir des propriétés de message IBM MQ sur le premier message d'un groupe en spécifiant l'agent à utiliser comme destination du transfert de fichiers et le nom de fichier vers lequel transférer les données.

Pour plus d'informations, voir la section [«Surveillance d'une file d'attente et utilisation de la substitution de variable»](#), à la page 191.

Conseils pour la définition des attributs MQ et des propriétés MFT associées à la taille des messages

Vous pouvez modifier les attributs IBM MQ et les propriétés Managed File Transfer pour affecter le comportement de Managed File Transfer lors de la lecture ou de l'écriture de messages de différentes tailles.

Si la taille des messages lus à partir d'une file d'attente source ou écrits dans une file d'attente de destination dépasse 1048576 octets (1 Mo), vous devez augmenter la valeur de la propriété Managed File Transfer Agent **maxInputOutputMessageLength** à une valeur supérieure ou égale à la taille maximale des messages à lire ou à écrire.

Si les messages de la file d'attente source sont supérieurs à 1048576 octets, vous devez définir la propriété **maxInputOutputMessageLength** sur l'agent source. Si les messages de la file d'attente de destination sont supérieurs à 1048576 octets, vous devez définir la propriété **maxInputOutputMessageLength** sur l'agent de destination. Pour plus d'informations sur la propriété **maxInputOutputMessageLength**, voir [Propriétés avancées de l'agent](#).

- Si la file d'attente dans laquelle l'agent est en cours d'écriture ou de lecture est locale dans le gestionnaire de files d'attente de l'agent, vous devrez peut-être modifier les attributs du IBM MQ gestionnaire de files d'attente, de la file d'attente et du canal **MAXMSGL**.

Vérifiez que la valeur de la taille de message maximale de la file d'attente source ou de destination est supérieure ou égale à la valeur de la propriété d'agent **maxInputOutputMessageLength**.

Vérifiez que la valeur de chacun des attributs IBM MQ suivants, en octets:

- Taille maximale des messages du gestionnaire de files d'attente de l'agent
- Taille maximale des messages du système `SYSTEM.FTE.STATE.nom_agent` file d'attente
- Taille maximale des messages du canal client, si votre agent se connecte au gestionnaire de files d'attente en mode client

est supérieur ou égal au résultat du calcul suivant:

Pour un transfert de fichier à message (qui prend en charge une taille de fichier pouvant atteindre 100 Mo):

La valeur de **maxInputOutputMessageLength**

Pour un transfert de message vers fichier:

La valeur de $3 * (\text{maxInputOutputMessageLength}) + 1048576$

(Ce calcul est dérivé du fait que trois points de contrôle peuvent être stockés dans un message d'état et que chaque point de contrôle peut avoir à mettre en mémoire tampon jusqu'à la taille maximale d'une quantité de données de message.)

- Si la file d'attente dans laquelle l'agent écrit est une file d'attente éloignée, vous devrez peut-être modifier les attributs **MAXMSGL** du gestionnaire de files d'attente, de la file d'attente et du canal IBM MQ.

Vérifiez que la valeur de chacun des attributs IBM MQ suivants est supérieure ou égale à la valeur de la propriété d'agent **maxInputOutputMessageLength**:

- Taille maximale des messages de la file d'attente de transmission du gestionnaire de files d'attente éloignées sur le gestionnaire de files d'attente de l'agent
- Taille maximale des messages du canal entre le gestionnaire de files d'attente de l'agent et le gestionnaire de files d'attente éloignées
- Taille maximale des messages de la file d'attente de destination sur le gestionnaire de files d'attente éloignées
- Taille maximale des messages du gestionnaire de files d'attente éloignées

Vérifiez que la valeur de chacun des attributs IBM MQ suivants, en octets:

- Taille maximale des messages du gestionnaire de files d'attente de l'agent
- Taille maximale des messages du système `SYSTEM.FTE.STATE.nom_agent` file d'attente
- Taille maximale des messages du canal client, si votre agent se connecte au gestionnaire de files d'attente en mode client

est supérieur ou égal au résultat du calcul suivant:

Pour un transfert de fichier à message (qui prend en charge une taille de fichier pouvant atteindre 100 Mo):

La valeur de **maxInputOutputMessageLength**

Pour un transfert de message vers fichier:

La valeur de $3 * (\text{maxInputOutputMessageLength}) + 1048576$

(Ce calcul est dérivé du fait que trois points de contrôle peuvent être stockés dans un message d'état et que chaque point de contrôle peut avoir à mettre en mémoire tampon jusqu'à la taille maximale d'une quantité de données de message.)

Si vous dépassez la valeur de l'une de ces propriétés, l'agent s'arrête avec l'erreur suivante dans le journal des événements de l'agent:

```
BFGUT0002E: An internal error has occurred. Product failure data was captured in file
"FFDC.FTE.20100928170828514.8172766022149157013.log".
BFGSS0025E: An internal error has occurred. The exception is: cc=2 rc=2010 op=put - MQPUT to
SYSTEM.FTE.STATE.agent_name
BFGAG0061E: The agent ended abnormally
```

Les codes anomalie IBM MQ suivants peuvent être inclus dans ce message dans le journal des événements de l'agent:

- **rc=2010** Ce code anomalie est mappé à **MQRC_DATA_LENGTH_ERROR** et indique que la valeur de la taille maximale de message du canal client a été dépassée. Pour résoudre ce problème, assurez-vous que la taille de message maximale du canal client du gestionnaire de files d'attente de l'agent est supérieure ou égale au résultat du calcul suivant:

```
3 * (maxInputOutputMessageLength) + 1048576
```

- **rc=2030** Ce code anomalie est mappé à **MQRC_MSG_TOO_BIG_FOR_Q** et indique la valeur de la taille de message maximale de **SYSTEM.FTE.STATE.nom_agent** a été dépassée. Pour résoudre ce problème, assurez-vous que la taille de message maximale de **SYSTEM.FTE.STATE**. La file d'attente **nom_agent** est supérieure ou égale au résultat du calcul suivant:

```
3 * (maxInputOutputMessageLength) + 1048576
```

- **rc=2031** Ce code anomalie est mappé à **MQRC_MSG_TOO_BIG_FOR_Q_MGR** et indique que la valeur de la taille de message maximale du gestionnaire de files d'attente de l'agent a été dépassée. Pour résoudre ce problème, assurez-vous que la taille de message maximale du gestionnaire de files d'attente de l'agent est supérieure ou égale au résultat du calcul suivant:

```
3 * (maxInputOutputMessageLength) + 1048576
```

Si vous transférez de nombreux messages de petite taille

Si la taille moyenne des messages que l'agent lit ou écrit dans une file d'attente est inférieure à 1310 octets et que l'agent lit ou écrit plus de 10000 messages, vous devez augmenter le nombre maximal d'attributs de messages non validés sur le gestionnaire de files d'attente ou réduire la quantité de données dans un intervalle de point de contrôle.

Lorsque l'agent lit ou écrit des messages dans une file d'attente, les **GET** ou **PUT** correspondants sont regroupés dans des transactions. Le nombre de **GETs** ou de **PUTs** dans une transaction est déterminé par le nombre requis pour traiter toutes les données dans un intervalle de point de contrôle. La quantité approximative des données dans un intervalle de point de contrôle est déterminée à partir des propriétés de l'agent à l'aide du calcul suivant:

```
Checkpoint interval data size (in bytes) = agentCheckpointInterval * agentFrameSize *
agentWindowSize * agentChunkSize.
```

La taille des données de point de contrôle par défaut est $1 * 5 * 10 * 262144$ octets = 13107200 octets (12.5MB). Le nombre maximal de messages non validés dans une transaction pris en charge par un gestionnaire de files d'attente est contrôlé par l'attribut de gestionnaire de files d'attente **MaxUncommittedMsgs**. La valeur par défaut de cet attribut est 10000 messages. Si la taille moyenne

des messages est inférieure à environ 1310 octets, le nombre maximal par défaut de messages non validés est dépassé s'il y a plus de 10000 messages à écrire.

Si vous dépassez la limite **MaxUncommittedMsgs**, l'agent s'arrête avec l'erreur suivante dans le journal des événements de l'agent:

```
BFGSS0024E: The agent has received a reason code of '2024' from the message queue interface (MQI).  
The agent cannot continue processing and will now end.  
BFGAG00139I: The agent has suspended its current transfers and is now stopping.
```

Le code anomalie 2024 est mappé à: MQRC_SYNCPOINT_LIMIT_ATTEINTES.

Pour résoudre ce problème, effectuez l'une des actions suivantes:

- Augmentez la valeur de l'attribut de gestionnaire de files d'attente **MaxUncommittedMsgs** du gestionnaire de files d'attente auquel se connecte l'agent lors de la lecture ou de l'écriture dans une file d'attente. Voir [MaxUncommittedMsgs \(MQLONG\)](#).
- Réduisez la quantité de données dans un intervalle de point de contrôle. Pour ce faire, diminuez la valeur d'une ou de plusieurs des propriétés d'agent suivantes:
 - agentCheckpointInterval
 - agentFrameSize
 - agentWindowSize
 - agentChunkSize

Pour plus d'informations sur ces propriétés d'agent, voir [Propriétés d'agent avancées](#).

Si vous écrivez des messages dans une file d'attente de manière persistante

Si vous effectuez un transfert vers une file d'attente et que vous écrivez les messages dans la file d'attente de manière persistante, vous devrez peut-être augmenter la taille de l'espace fichier journal du gestionnaire de files d'attente pour pouvoir consigner toutes les données dans un intervalle de point de contrôle.

Si vous dépassez l'espace fichier journal du gestionnaire de files d'attente, l'agent s'arrête avec l'erreur suivante dans le journal des événements de l'agent:

```
BFGSS0024E: The agent has received a reason code of '2102' from the message queue interface (MQI).  
The agent cannot continue processing and will now end.  
BFGAG0062E: The agent has received MQI reason code '2102'. The agent cannot continue processing and  
will now end.  
BFGAG0061E: The agent ended abnormally
```

Le code anomalie '2102' est mappé à: MQRC_RESOURCE_PROBLEM.

Pour résoudre ce problème, augmentez la taille de l'espace fichier journal du gestionnaire de files d'attente de l'agent cible.

Conseils pour la spécification d'un temps d'attente sur un transfert de message vers fichier

Lorsque vous spécifiez un transfert de message à fichier, vous pouvez éventuellement spécifier un temps d'attente sur le transfert à l'aide du paramètre **-sqwt**. La valeur de **-sqwt** correspond à la durée pendant laquelle l'agent source attend qu'un message apparaisse dans la file d'attente source si la file d'attente source est vide ou devient vide, ou qu'un groupe complet apparaisse dans la file d'attente source si l'attribut **-sqgi** est spécifié.

Cette rubrique décrit les paramètres utilisés dans la commande **fteCreateTransfer** pour spécifier un temps d'attente. Vous pouvez également spécifier le temps d'attente à l'aide de la valeur [srcqueuetimeout](#) du paramètre **fte:filespec**.

Si la valeur du paramètre **-sqwt** est supérieure ou égale à la durée pendant laquelle l'agent de destination attend que le transfert soit terminé par l'agent source, le transfert n'est pas terminé. La durée pendant laquelle l'agent de destination attend la fin du transfert est calculée comme suit:

```
transferAckTimeout * transferAckTimeoutRetries
```

Les propriétés `transferAckTimeout` et `transferAckTimeoutRetries` sont définies dans le fichier `agent.properties` de l'agent de destination. Pour plus d'informations sur ces propriétés d'agent, voir «Le fichier MFT `agent.properties`», à la page 564.

Pour éviter l'échec des transferts, vous devez effectuer l'une des étapes suivantes:

- Réduisez la valeur du paramètre **-sqwt** de sorte qu'elle soit inférieure à la valeur de la propriété `transferAckTimeout` de l'agent de destination.

Remarque : La valeur par défaut de la propriété `transferAckTimeout` est 60 000 millisecondes. La valeur du paramètre **-sqwt** est exprimée en secondes. La valeur est inférieure ou égale à 59.

- Augmentez la valeur de la propriété `transferAckTimeout` de l'agent de destination de sorte qu'elle soit supérieure à la valeur du paramètre **-sqwt**.

Remarque : La valeur de la propriété `transferAckTimeout` est exprimée en millisecondes. La valeur du paramètre **-sqwt** est exprimée en secondes.

Référence associée

«`fteCreateTransfer`: démarrer un nouveau transfert de fichier», à la page 447

La commande **`fteCreateTransfer`** crée et démarre un nouveau transfert de fichier à partir de la ligne de commande. Cette commande peut démarrer un transfert de fichier immédiatement, planifier un transfert de fichier à une heure et une date ultérieures, répéter un transfert planifié une ou plusieurs fois et déclencher un transfert de fichiers sous certaines conditions.

«Le fichier MFT `agent.properties`», à la page 564

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.

«`fte`: élément imbriqué `Ant filespec`», à la page 887

Le paramètre **`fte:filespec`** est utilisé en tant qu'élément imbriqué dans d'autres tâches. Utilisez

`fte:filespec` pour décrire un mappage entre un ou plusieurs fichiers source, répertoires  ou ensembles de données et une destination. Généralement, cet élément est utilisé lors de l'expression d'un ensemble de fichiers ou de répertoires  ou d'ensembles de données à déplacer ou à copier.

Pages de codes disponibles pour MFT

Cette rubrique de référence répertorie tous les formats de codage de caractères disponibles pour la conversion de fichier texte sur les différentes plateformes prises en charge par Managed File Transfer.

Codages communs

Ces formats de codage de caractères sont disponibles sur toutes les plateformes prises en charge. Si votre fichier source est codé à l'aide de l'un des formats de cette table et que vous souhaitez utiliser un autre format de cette table pour écrire le fichier cible, vous pouvez le faire sans tenir compte de la plateforme. Vous pouvez utiliser le nom canonique ou l'un des alias pour spécifier un format de codage.

Nom canonique	Alias
windows-1256	ibm-1256, Cp1256
windows-1255	ibm-1255, Cp1255
windows-1254	Cp1254, ibm-1254

Nom canonique	Alias
windows-1253	Cp1253, ibm-1253
windows-1252	ibm-1252, Cp1252
windows-1251	ibm-1251, Cp1251
windows-1250	Cp1250, ibm-1250
UTF-8	UTF_8, UTF8
UTF-16LE	X-UTF-16LE, UTF16LE, UTF_16LE, UnicodeLittlenon marqué
UTF-16BE	UTF16BE, UnicodeBignon marqué, ISO-10646-UCS-2, UTF_16BE, X-UTF-16BE
US-ASCII	Cp367, iso-ir-6, ANSI_X3.4-1968, ANSI_X3.4-1986, par défaut, ASCII, us, iso-646.irv:1983, csASCII, 646, ascii7, ISO646-US, ibm-367, ISO-646.irv:1991, direct
TIS-620	tis620, tis620.2533
IBM-1122	Cp1122, ibm1122
IBM-1006	Cp1006, ibm1006
IBM-037	ibm-37
GB18030	windows-54936, gb18030-2000, ibm-1392
EUC-TW	x-euc-tw, euctw, cns11643, euc_tw
EUC-KR	ibm-euckr, euc_kr, ksc_5601, ks_c_5601-1987, ksc5601_1987, euckr, ksc5601-1987, ibm-970, Cp970, 5601
EUC-JP	x-euc-jp, euc_jp, eucjp, x-eucjp, euc_jp_linux, euc-jp-linux
EUC-CN	x-euc-cn, ibm-euccn, euc_cn, euccn
Big5	big5-0, big5, Big5-HKSCS
IBM-1025	Cp1025, ibm1025
IBM-1026	ibm1026, Cp1026
IBM-1046	Cp1046, ibm1046
IBM-1097	Cp1097, ibm1097
IBM-1098	Cp1098, ibm1098
IBM-1112	ibm1112, Cp1112
IBM-1383	Cp1383, ibm1383
IBM-273	Cp273, ibm273
IBM-277	Cp277, ibm277
IBM-278	Cp278, ibm278
IBM-280	ibm280, Cp280
IBM-284	ibm284, Cp284
IBM-285	Cp285, ibm285
IBM-297	ibm297, Cp297
IBM-420	Cp420, ibm420
IBM-860	Cp860, ibm860
IBM-861	ibm861, Cp861
IBM-862	Cp862, ibm862
IBM-863	Cp863, ibm863
IBM-864	Cp864, ibm864
IBM-865	ibm865, Cp865

Nom canonique	Alias
windows-1257	Cp1257, ibm-1257
windows-1258	Cp1258, ibm-1129, ibm-1258
windows-31j	ms_kanji, cswindows31j, MS932, windows-932
windows-874	MS874
windows-936	MS936, x-mswin-936, 936
windows-949	MS949, Cp1361, ibm-1361, ibm1361, ms1361, ksc5601-1992, x-windows-949
windows-950	MS950, x-windows-950
IBM-857	ibm857, Cp857, csibm857
IBM-856	Cp856, ibm856
IBM-855	Cp855, ibm855
IBM-852	cspcp852, ibm852, Cp852
IBM-850	Cp850, ibm850, cspc850multilingual
IBM-838	Cp838, ibm838
IBM-834	Cp834, ibm834
IBM-775	ibm775, Cp775
IBM-737	Cp737, ibm737
IBM-500	Cp500, ibm500
IBM-437	ibm437, Cp437, cspc8codepage437
IBM-424	ibm424, Cp424
IBM-1123	Cp1123, ibm1123
IBM-1124	Cp1124, ibm1124
IBM-1381	Cp1381, ibm1381
IBM-866	Cp866, ibm866
IBM-868	Cp868, ibm868
IBM-869	ibm869, Cp869
IBM-870	Cp870, ibm870
IBM-871	ibm871, Cp871
IBM-874	ibm874, Cp874
IBM-875	Cp875, ibm875
IBM-921	Cp921, ibm921
IBM-922	Cp922, ibm922
IBM-933	Cp933, ibm933
IBM-935	Cp935, ibm935
IBM-937	Cp937, ibm937
IBM-942	Cp942, ibm942
IBM-943	Cp943, ibm943
IBM-948	ibm948, Cp948
IBM-949	ibm949, Cp949
IBM-950	ibm950, Cp950
ISCI91	ISCII

Nom canonique	Alias
ISO-2022-CN	iso2022-cn-cns, iso2022cn-cns, iso-2022-cn-cns, iso2022cn, iso2022-cn
ISO-2022-CN-GB	iso2022-cn-gb, iso2022cn-gb
ISO-2022-JP	iso2022jp, jis, iso2022-jp, iso-2022-jp2, csiso2022jp2, csjisencoding, jis-encoding
ISO-2022-KR	csiso2022kr, iso2022-kr, iso2022kr
ISO-8859-1	iso8859_1, iso8859-1, ibm819, l1, csisolatin1, Cp819, iso-ir-100, iso-8859-1:1987, ibm-819, latin1, 8859-1
ISO-8859-13	iso8859-13, 8859-13, iso8859_13
ISO-8859-15	csisolatin9, iso8859-15, ibm923, latin9, ibm-923, l9, iso8859_15, iso8859_15_fdis, Cp923, latin0
ISO-8859-2	Cp912, ibm912, iso8859-2, iso-8859-2:1987, l2, iso8859_2, csisolatin2, latin2, ibm-912, 8859-2, iso-ir-101
ISO-8859-3	iso8859-3, Cp913, l3, iso8859_3, iso-ir-109, iso-8859-3:1988, latin3, ibm-913, 8859-3, csisolatin3
ISO-8859-4	Cp914, latin4, iso8859_4, l4, iso-8859-4:1988, ibm-914, iso8859-4, 8859-4, csisolatin4, iso-ir-110
ISO-8859-5	csisolatincyrillique, iso-ir-144, cyrillique, iso8859_5, iso-8859-5:1988, ibm-915, 8859-5, Cp915, ibm915, iso8859-5
ISO-8859-6	csisolatinarabic, Cp1089, iso-8859-6:1987, ecma-114, iso-ir-127, asmo-708, iso8859_6, 8859-6, ibm1089, arabic, iso8859-6, ibm-1089
ISO-8859-7	ecma-118, ibm813, csisolatingreek, elot-928, iso-ir-126, Cp813, 8859-7, iso-8859-7:1987, iso8859_7, greek, greek8, ibm-813, iso8859-7
ISO-8859-8	iso-ir-138, iso-8859-8:1988, csisolatinhebrew, hebrew, iso8859-8, 8859-8, ibm-916, iso8859_8, Cp916, ibm916
ISO-8859-9	ibm-920, ibm920, latin5, 8859-9, Cp920, l5, iso8859-9, iso8859_9, csisolatin5, iso-ir-148
JIS0212	
KOI8-R	koi8, ibm-878, cskoi8r, koi8_r
MacArabic	
MacCentralEurope	ibm-1282
MacCroatian	ibm-1284
MacCyrillic	ibm-1283
MacGreek	ibm-1280
MacIceland	ibm-1286
MacRoman	ibm-1275
MacRomania	ibm-1285
MacSymbol	Adobe-Symbol-Encoding, ibm-1038
MacTurkish	ibm-1281

Codages par défaut de la plateforme source

Si vous ne spécifiez pas de codage pour le fichier source ou pour le fichier cible, le codage par défaut de cette plateforme sera utilisé. La conversion est effectuée par l'agent de destination et les codages source et de destination doivent être pris en charge sur la plateforme de l'agent de destination pour que la conversion ait lieu. Le codage par défaut de la destination sera toujours pris en charge sur l'agent de destination, il est donc toujours sûr de ne pas le spécifier. Toutefois, il peut ne pas être sûr d'utiliser un codage source par défaut, car l'agent de destination peut ne pas prendre en charge la valeur par défaut de la source.

Si vous utilisez des codages de source par défaut, vous devez utiliser les tableaux de cette rubrique pour vous assurer que la combinaison est prise en charge.

Plateforme	Codage par défaut
Solaris	ISO-8859-1
SUSE Linux Enterprise Server sous System x	UTF-8
 IBM i	ISO-8859-1
HP-UX (Itanium)	ISO-8859-1
Linux pour IBM Z	UTF-8
AIX	ISO-8859-1
Windows	windows-1252
Red Hat Enterprise Linux sur System x	UTF-8
 z/OS	IBM-1047
Linux on POWER Systems - Big Endian	UTF-8
HP (PA-RISC)	ISO-8859-1

Codages spécifiques à la plateforme

Remarque: Les deux tableaux suivants contiennent les mêmes informations. Il est organisé de deux manières différentes pour vous aider à trouver les informations correctes, selon que vous recherchez par plateforme ou par codage.

Codages par plateforme

Les noms canoniques sont répertoriés en gras, suivis des alias entre parenthèses.

Les plateformes qui prennent en charge uniquement les codages déjà répertoriés dans la table des codages communs ne sont pas répertoriées ici.

Plateforme	Codages pris en charge (pas dans la table des codages communs)
Solaris	<p> x-IBM33722 (ibm33722, 33722, ibm-33722_vascii_vpua, ibm-5050, ibm-33722, cp33722) x-IBM930 (cp930, ibm930, ibm-930, 930) x-IBM939 (ibm-939, ibm939, cp939, 939) x-IBM964 (964, cp964, ibm-964, ibm964) x-ISO-2022-CN-CNS (ISO-2022-CN-CNS, ISO2022CN_CNS) x-iso-8859-11 (iso-8859-11, iso8859_11) x-JISAutoDetect (JISAutoDetect) x-MS932_0213 () x-MS950-HKSCS (MS950_HKSCS) x-PCK (pck) x-SJIS_0213 () X-UTF-32BE-BOM (UTF_32BE_BOM, UTF-32BE-BOM) x-MacUkraine (macukraine) x-MacThai (macthai) x-MacHebrew (machebrew) x-MacDingbat (macdingbat) x-KSC5601 (ksc5601) x-JIS0208 (jis_c6226-1983, jis_x0208-1983, csiso87jisx0208, x0208, iso-ir-87, jis0208) x-IBM949C (ibm949c, cp949c, 949c, ibm-949c) x-IBM943C (cp943c, 943c, ibm-943c, ibm943c) JIS_X0201 (jis_x0201, x0201, cshalfwidthkatakana, jis0201) x-windows-iso2022jp (windows-iso2022jp) x-windows-50221 (ms50221, cp50221) x-windows-50220 (cp50220, ms50220) X-UTF-32LE-BOM (UTF_32LE_BOM, UTF-32LE-BOM) x-eucJP-Open (EUC_JP_Solaris, eucJP-open) x-Big5-Solaris (Big5_Solaris) ISO-2022-JP-2 (csISO2022JP2, iso2022jp2) IBM918 (cp918, ebcdic-cp-ar2, ibm-918, 918) IBM1047 (cp1047, 1047, ibm-1047) IBM01149 (cp1149, cp01149, ccsid01149, 1149) IBM01148 (cp1148, ccsid01148, 1148, cp01148) IBM01147 (ccsid01147, cp1147, 1147, cp01147) IBM01146 (ccsid01146, cp01146, cp1146, 1146) IBM01145 (cp1145, cp01145, ccsid01145, 1145) IBM01144 (cp01144, cp1144, ccsid01144, 1144) IBM01143 (cp01143, 1143, ccsid01143, cp1143) IBM01142 (cp01142, cp1142, 1142, ccsid01142) IBM01141 (cp1141, ccsid01141, cp01141, 1141) IBM01140 (ccsid01140, cp01140, 1140, cp1140) IBM00858 (cp858, ccsid00858, 858, cp00858) X-UnicodeLittle (UnicodeLittle) X-UnicodeBig (UnicodeBig) COMPOUND_TEXT (x-texte-composé, x11-compound-text) IBM-942C (Cp942C, ibm942C) KOI8-U (koi8_u, ibm-1167) UTF-32 (UCS-4, UTF32, ISO-10646-UCS-4) UTF-32BE (UTF_32BE, X-UTF-32BE, UTF32BE) UTF-32LE (UTF_32LE, X-UTF-32LE, UTF32LE) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
SUSE Linux Enterprise Server sous System x	<p> windows-1256S (Cp1256s, ibm-1256s) UTF-8J (UTF8J) UTF-32LE (UTF_32LE, X-UTF-32LE, UTF32LE) UTF-32BE (UTF_32BE, X-UTF-32BE, UTF32BE) UTF-32 (UCS-4, UTF32, ISO-10646-UCS-4) PTCP154 (PT154, IBM-1169, Cyrillique-Asiatique, csPTCP154) KOI8-RU (ibm-1168, koi8_ru) ISO-8859-16 (8859-16, iso8859_16, iso8859-16) ISO-8859-14 (ISO_8859-14:1998, 8859-14, latin8, iso-ir-199, iso8859-14, l8, isoceltic, iso8859_14) IBM01141 (cp1141, ccsid01141, cp01141, 1141) IBM01142 (cp01142, cp1142, 1142, ccsid01142) IBM01143 (cp01143, 1143, ccsid01143, cp1143) IBM01144 (cp01144, cp1144, ccsid01144, 1144) IBM01145 (cp1145, cp01145, ccsid01145, 1145) IBM01146 (ccsid01146, cp01146, cp1146, 1146) IBM01147 (ccsid01147, cp1147, 1147, cp01147) IBM01148 (cp1148, ccsid01148, 1148, cp01148) IBM01149 (cp1149, cp01149, ccsid01149, 1149) IBM1047 (cp1047, 1047, ibm-1047) IBM918 (cp918, ebcdic-cp-ar2, ibm-918, 918) ISO-2022-JP-2 (csISO2022JP2, iso2022jp2) x-Big5-Solaris (Big5_Solaris) x-eucJP-Open (EUC_JP_Solaris, eucJP-open) x-IBM33722 (ibm33722, 33722, ibm-33722_vascii_vpua, ibm-5050, ibm-33722, cp33722) x-IBM930 (cp930, ibm930, ibm-930, 930) x-IBM939 (ibm-939, ibm939, cp939, 939) x-IBM964 (964, cp964, ibm-964, ibm964) x-ISO-2022-CN-CNS (ISO-2022-CN-CNS, ISO2022CN_CNS) x-iso-8859-11 (iso-8859-11, iso8859_11) x-JISAutoDetect (JISAutoDetect) x-MS932_0213 () x-MS950-HKSCS (MS950_HKSCS) x-PCK (pck) x-IBM1363C (ibm1363c, cp1363c, ibm-1363c) x-IBM420S (420s, ibm-420s, csibm420s, ibm420s, cp420s) x-IBM864S (csibm864s, ibm864s, cp864s, 864s, ibm-864s) x-IBM943C (cp943c, 943c, ibm-943c, ibm943c) x-IBM949C (ibm949c, cp949c, 949c, ibm-949c) x-IBM954C (cp954c, 954c, ibm-954c, ibm954c) x-ISO-8859-6S (8859_6s, iso8859-6s, iso8859_6s, iso-8859-6s) x-JIS0208 (jis_c6226-1983, jis_x0208-1983, csiso87jisx0208, x0208, iso-ir-87, jis0208) x-KSC5601 (ksc5601) x-MacDingbat (macdingbat) x-MacHebrew (machebrew) x-MacThai (macthai) x-MacUkraine (macukraine) x-IBM1046S (ibm-1046s, 1046s, cp1046s, ibm1046s) x-IBM-udcJP (IBM-udcJP) JIS_X0201 (jis_x0201, x0201, cshalfwidthkatakana, jis0201) IBM-939A (Cp939A, ibm939A) IBM-930A (ibm930A, Cp930A) IBM-33722A (Cp33722A, ibm33722A) x-windows-iso2022jp (windows-iso2022jp) x-windows-50221 (ms50221, cp50221) x-windows-50220 (cp50220, ms50220) X-UTF-32LE-BOM (UTF_32LE_BOM, UTF-32LE-BOM) X-UTF-32BE-BOM (UTF_32BE_BOM, UTF-32BE-BOM) x-SJIS_0213 () IBM01140 (ccsid01140, cp01140, 1140, cp1140) IBM00858 (cp858, ccsid00858, 858, cp00858) X-UnicodeLittle (UnicodeLittle) X-UnicodeBig (UnicodeBig) IBM-859 (Cp859, ibm859) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
SUSE Linux Enterprise Server sous System x	<p> IBM-837 (ibm837, Cp837) IBM-836 (ibm836, Cp836) IBM-835 (ibm835, Cp835) IBM-833 (ibm833, Cp833) IBM-808 (Cp808, ibm808) IBM-720 (Cp720, ibm720) IBM-33722C (ibm-eucjp, Cp33722c) IBM-301 (Cp301, ibm301) IBM-300 (Cp300, ibm300) IBM-290 (ibm290, Cp290) IBM-1399 (ibm1399, Cp1399) IBM-1390 (Cp1390, ibm1390) IBM-1388 (Cp1388, ibm1388) IBM-1385 (Cp1385, ibm1385) IBM-1382 (ibm1382, Cp1382) IBM-1088 (Cp1088, ibm1088) IBM-1043 (Cp1043, ibm1043) IBM-1041 (Cp1041, ibm1041) IBM-1027 (Cp1027, ibm1027) CESU-8 (CESU8) COMPOUND_TEXT (x-texte-composé, x11-compound-text) GB2312 (gb2312-1980, gb2312-80) GBK (GBK) hp-roman8 (roman8, ibm-1051, r8, Cp1051) IBM-1114 (Cp1114, ibm1114) IBM-1115 (Cp1115, ibm1115) IBM-1351 (Cp1351, ibm1351) IBM-1362 (Cp1362, ibm1362) IBM-1363 (ibm1363, Cp1363) IBM-1364 (Cp1364, ibm1364) IBM-1370 (Cp1370, ibm1370) IBM-1371 (Cp1371, ibm1371) IBM-1380 (Cp1380, ibm1380) IBM-867 (Cp867, ibm867) IBM-897 (Cp897, ibm897) IBM-924 (Cp924, ibm924) IBM-927 (ibm927, Cp927) IBM-932 (ibm932, Cp932) IBM-947 (Cp947, ibm947) IBM-951 (Cp951, ibm951) IBM-954 (ibm954, Cp954) IBM-971 (Cp971, ibm971) ISO-8859-10 (latin6, 8859-10, ISO_8859-10:1992, iso8859_10, iso-ir-157, ibm-919, iso8859-10, l6, csisolatin6) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
<p>IBM i IBM i IBM i</p>	<p> windows-1256S (Cp1256s, ibm-1256s) UTF-8J (UTF8J) IBM-1146 (Cp1146, ibm1146) IBM-1145 (Cp1145, ibm1145) IBM-1144 (ibm1144, Cp1144) IBM-1143 (Cp1143, ibm1143) IBM-1142 (Cp1142, ibm1142) IBM-1141 (Cp1141, ibm1141) IBM-1140 (ibm1140, Cp1140) IBM-1115 (Cp1115, ibm1115) IBM-1114 (Cp1114, ibm1114) hp-roman8 (roman8, ibm-1051, r8, Cp1051) GBK (GBK) GB2312 (gb2312-1980, gb2312-80) COMPOUND_TEXT (x-texte-composé, x11-compound-text) CESU-8 (CESU8) IBM-1027 (Cp1027, ibm1027) IBM-1041 (Cp1041, ibm1041) IBM-1043 (Cp1043, ibm1043) IBM-1046S (ibm1046S, Cp1046S) IBM-1047 (Cp1047, ibm1047) IBM-1088 (Cp1088, ibm1088) IBM-1382 (ibm1382, Cp1382) IBM-1385 (Cp1385, ibm1385) IBM-1386 (ibm1386, Cp1386) IBM-1388 (Cp1388, ibm1388) IBM-836 (ibm836, Cp836) IBM-837 (ibm837, Cp837) IBM-858 (Cp858, ibm858) IBM-859 (Cp859, ibm859) IBM-864S (ibm864S, Cp864S) X-UnicodeBig (UnicodeBig) X-UnicodeLittle (UnicodeLittle) IBM-1047_LF (Cp1047_LF, ibm1047_LF) IBM-1141_LF (Cp1141_LF, ibm1141_LF) IBM-33722A (Cp33722A, ibm33722A) IBM-924_LF (Cp924_LF, ibm924_LF) IBM-930A (ibm930A, Cp930A) IBM-939A (Cp939A, ibm939A) IBM-835 (ibm835, Cp835) IBM-833 (ibm833, Cp833) IBM-808 (Cp808, ibm808) IBM-720 (Cp720, ibm720) IBM-420S (Cp420S, ibm420S) IBM-33722C (ibm-eucjp, Cp33722c) IBM-33722 (5050, Cp5050) IBM-301 (Cp301, ibm301) IBM-300 (Cp300, ibm300) IBM-290 (ibm290, Cp290) IBM-1399 (ibm1399, Cp1399) IBM-1390 (Cp1390, ibm1390) IBM-1147 (Cp1147, ibm1147) IBM-1148 (ibm1148, Cp1148) IBM-1149 (Cp1149, ibm1149) IBM-1351 (Cp1351, ibm1351) IBM-1362 (Cp1362, ibm1362) IBM-1363 (ibm1363, Cp1363) IBM-1363C (ibm1363C, Cp1363C) IBM-1364 (Cp1364, ibm1364) IBM-1370 (Cp1370, ibm1370) IBM-1371 (Cp1371, ibm1371) IBM-1380 (Cp1380, ibm1380) IBM-867 (Cp867, ibm867) IBM-897 (Cp897, ibm897) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
  IBM i	<p> IBM-918 (ibm918, Cp918) IBM-924 (Cp924, ibm924) IBM-927 (ibm927, Cp927) IBM-930 (Cp5026, 5026) IBM-932 (ibm932, Cp932) IBM-939 (Cp5035, 5035) IBM-942C (Cp942C, ibm942C) IBM-943C (ibm943C, Cp943C) IBM-947 (Cp947, ibm947) IBM-949C (Cp949C, ibm949C) IBM-951 (Cp951, ibm951) IBM-954 (ibm954, Cp954) IBM-954C (Cp954c) IBM-964 (ibm-euctw, Cp964) IBM-971 (Cp971, ibm971) ISO-8859-10 (latin6, 8859-10, ISO_8859-10:1992, iso8859_10, iso-ir-157, ibm-919, iso8859-10, l6, csisolatin6) ISO-8859-14 (ISO_8859-14:1998, 8859-14, latin8, iso-ir-199, iso8859-14, l8, isoceltic, iso8859_14) ISO-8859-16 (8859-16, iso8859_16, iso8859-16) ISO-8859-6S (iso8859-6S, iso8859_6S) JIS0201 () JIS0208 () Johab (x-johab) KOI8-RU (ibm-1168, koi8_ru) KOI8-U (koi8_u, ibm-1167) KSC5601 () MacDingbat () MacHebrew () MacThai () MacUkraine () PTCP154 (PT154, IBM-1169, Cyrillique-Asiatique, csPTCP154) Shift_JIS () UTF-16 (UTF16, Unicode, UTF_16, UCS-2) UTF-32 (UCS-4, UTF32, ISO-10646-UCS-4) UTF-32BE (UTF_32BE, X-UTF-32BE, UTF32BE) UTF-32LE (UTF_32LE, X-UTF-32LE, UTF32LE) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
HP-UX (Itanium)	<p> UTF-16 (UTF16, Unicode, UTF_16, UCS-2) MacUkraine () MacThai () MacHebrew () MacDingbat () JISO208 () JISO201 () IBM-949C (Cp949C, ibm949C) IBM-943C (ibm943C, Cp943C) IBM-942C (Cp942C, ibm942C) IBM00858 (cp858, ccsid00858, 858, cp00858) IBM01140 (ccsid01140, cp01140, 1140, cp1140) x-eucJP-Open (EUC_JP_Solaris, eucJP-open) x-IBM33722 (ibm33722, 33722, ibm-33722_vascii_vpua, ibm-5050, ibm-33722, cp33722) x-IBM930 (cp930, ibm930, ibm-930, 930) x-IBM939 (ibm-939, ibm939, cp939, 939) x-IBM964 (964, cp964, ibm-964, ibm964) x-ISO-2022-CN-CNS (ISO-2022-CN-CNS, ISO2022CN_CNS) x-iso-8859-11 (iso-8859-11, iso8859_11) x-JISAutoDetect (JISAutoDetect) x-MS950-HKSCS (MS950_HKSCS) x-PCK (pck) x-windows-50220 (cp50220, ms50220) x-windows-50221 (ms50221, cp50221) x-windows-iso2022jp (windows-iso2022jp) x-Big5-Solaris (Big5_Solaris) IBM918 (cp918, ebcdic-cp-ar2, ibm-918, 918) IBM1047 (cp1047, 1047, ibm-1047) IBM01149 (cp1149, cp01149, ccsid01149, 1149) IBM01148 (cp1148, ccsid01148, 1148, cp01148) IBM01147 (ccsid01147, cp1147, 1147, cp01147) IBM01146 (ccsid01146, cp01146, cp1146, 1146) IBM01145 (cp1145, cp01145, ccsid01145, 1145) IBM01144 (cp01144, cp1144, ccsid01144, 1144) IBM01143 (cp01143, 1143, ccsid01143, cp1143) IBM01142 (cp01142, cp1142, 1142, ccsid01142) IBM01141 (cp1141, ccsid01141, cp01141, 1141) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
Linux pour IBM Z	<p> windows-1256S (Cp1256s, ibm-1256s) UTF-8J (UTF8J) UTF-32LE (UTF_32LE, X-UTF-32LE, UTF32LE) UTF-32BE (UTF_32BE, X-UTF-32BE, UTF32BE) UTF-32 (UCS-4, UTF32, ISO-10646-UCS-4) PTCP154 (PT154, IBM-1169, Cyrillique-Asiatique, csPTCP154) KOI8-RU (ibm-1168, koi8_ru) ISO-8859-16 (8859-16, iso8859_16, iso8859-16) ISO-8859-14 (ISO_8859-14:1998, 8859-14, latin8, iso-ir-199, iso8859-14, l8, isoceltique, iso8859_14) IBM01141 (cp1141, ccsid01141, cp01141, 1141) IBM01142 (cp01142, cp1142, 1142, ccsid01142) IBM01143 (cp01143, 1143, ccsid01143, cp1143) IBM01144 (cp01144, cp1144, ccsid01144, 1144) IBM01145 (cp1145, cp01145, ccsid01145, 1145) IBM01146 (ccsid01146, cp01146, cp1146, 1146) IBM01147 (ccsid01147, cp1147, 1147, cp01147) IBM01148 (cp1148, ccsid01148, 1148, cp01148) IBM01149 (cp1149, cp01149, ccsid01149, 1149) IBM1047 (cp1047, 1047, ibm-1047) IBM918 (cp918, ebcdic-cp-ar2, ibm-918, 918) ISO-2022-JP-2 (csISO2022JP2, iso2022jp2) x-Big5-Solaris (Big5_Solaris) x-eucJP-Open (EUC_JP_Solaris, eucJP-open) x-IBM33722 (ibm33722, 33722, ibm-33722_vascii_vpua, ibm-5050, ibm-33722, cp33722) x-IBM930 (cp930, ibm930, ibm-930, 930) x-IBM939 (ibm-939, ibm939, cp939, 939) x-IBM964 (964, cp964, ibm-964, ibm964) x-ISO-2022-CN-CNS (ISO-2022-CN-CNS, ISO2022CN_CNS) x-iso-8859-11 (iso-8859-11, iso8859_11) x-JISAutoDetect (JISAutoDetect) x-MS932_0213 () x-MS950-HKSCS (MS950_HKSCS) x-PCK (pck) x-IBM1363C (ibm1363c, cp1363c, ibm-1363c) x-IBM420S (420s, ibm-420s, csibm420s, ibm420s, cp420s) x-IBM864S (csibm864s, ibm864s, cp864s, 864s, ibm-864s) x-IBM943C (cp943c, 943c, ibm-943c, ibm943c) x-IBM949C (ibm949c, cp949c, 949c, ibm-949c) x-IBM954C (cp954c, 954c, ibm-954c, ibm954c) x-ISO-8859-6S (8859_6s, iso8859-6s, iso8859_6s, iso-8859-6s) x-JIS0208 (jis_c6226-1983, jis_x0208-1983, csiso87jisx0208, x0208, iso-ir-87, jis0208) x-KSC5601 (ksc5601) x-MacDingbat (macdingbat) x-MacHebrew (machebrew) x-MacThai (macthai) x-MacUkraine (macukraine) x-IBM1046S (ibm-1046s, 1046s, cp1046s, ibm1046s) x-IBM-udcJP (IBM-udcJP) JIS_X0201 (jis_x0201, x0201, cshalfwidthkatakana, jis0201) IBM-939A (Cp939A, ibm939A) IBM-930A (ibm930A, Cp930A) IBM-33722A (Cp33722A, ibm33722A) x-windows-iso2022jp (windows-iso2022jp) x-windows-50221 (ms50221, cp50221) x-windows-50220 (cp50220, ms50220) X-UTF-32LE-BOM (UTF_32LE_BOM, UTF-32LE-BOM) X-UTF-32BE-BOM (UTF_32BE_BOM, UTF-32BE-BOM) x-SJIS_0213 () IBM01140 (ccsid01140, cp01140, 1140, cp1140) IBM00858 (cp858, ccsid00858, 858, cp00858) X-UnicodeLittle (UnicodeLittle) X-UnicodeBig (UnicodeBig) IBM-859 (Cp859, ibm859) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
Linux pour IBM Z	<p> IBM-837 (ibm837, Cp837) IBM-836 (ibm836, Cp836) IBM-835 (ibm835, Cp835) IBM-833 (ibm833, Cp833) IBM-808 (Cp808, ibm808) IBM-720 (Cp720, ibm720) IBM-33722C (ibm-eucjp, Cp33722c) IBM-301 (Cp301, ibm301) IBM-300 (Cp300, ibm300) IBM-290 (ibm290, Cp290) IBM-1399 (ibm1399, Cp1399) IBM-1390 (Cp1390, ibm1390) IBM-1388 (Cp1388, ibm1388) IBM-1385 (Cp1385, ibm1385) IBM-1382 (ibm1382, Cp1382) IBM-1088 (Cp1088, ibm1088) IBM-1043 (Cp1043, ibm1043) IBM-1041 (Cp1041, ibm1041) IBM-1027 (Cp1027, ibm1027) CESU-8 (CESU8) COMPOUND_TEXT (x-texte-composé, x11-compound-text) GB2312 (gb2312-1980, gb2312-80) GBK (GBK) hp-roman8 (roman8, ibm-1051, r8, Cp1051) IBM-1114 (Cp1114, ibm1114) IBM-1115 (Cp1115, ibm1115) IBM-1351 (Cp1351, ibm1351) IBM-1362 (Cp1362, ibm1362) IBM-1363 (ibm1363, Cp1363) IBM-1364 (Cp1364, ibm1364) IBM-1370 (Cp1370, ibm1370) IBM-1371 (Cp1371, ibm1371) IBM-1380 (Cp1380, ibm1380) IBM-867 (Cp867, ibm867) IBM-897 (Cp897, ibm897) IBM-924 (Cp924, ibm924) IBM-927 (ibm927, Cp927) IBM-932 (ibm932, Cp932) IBM-947 (Cp947, ibm947) IBM-951 (Cp951, ibm951) IBM-954 (ibm954, Cp954) IBM-971 (Cp971, ibm971) ISO-8859-10 (latin6, 8859-10, ISO_8859-10:1992, iso8859_10, iso-ir-157, ibm-919, iso8859-10, l6, csisolatin6) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
AIX	<p> windows-1256S (Cp1256s, ibm-1256s) UTF-8J (UTF8J) UTF-32LE (UTF_32LE, X-UTF-32LE, UTF32LE) UTF-32BE (UTF_32BE, X-UTF-32BE, UTF32BE) UTF-32 (UCS-4, UTF32, ISO-10646-UCS-4) UTF-16 (UTF16, Unicode, UTF_16, UCS-2) Shift_JIS () PTCP154 (PT154, IBM-1169, Cyrillique-Asiatique, csPTCP154) MacUkraine () MacThai () MacHebrew () MacDingbat () KSC5601 () KOI8-U (koi8_u, ibm-1167) KOI8-RU (ibm-1168, koi8_ru) Johab (x-johab) JISO208 () JISO201 () ISO-8859-6S (iso8859-6S, iso8859_6S) ISO-8859-16 (8859-16, iso8859_16, iso8859-16) ISO-8859-14 (ISO_8859-14:1998, 8859-14, latin8, iso-ir-199, iso8859-14, l8, isoceltic, iso8859_14) ISO-8859-10 (latin6, 8859-10, ISO_8859-10:1992, iso8859_10, iso-ir-157, ibm-919, iso8859-10, l6, csisolatin6) IBM-971 (Cp971, ibm971) IBM-964 (ibm-euctw, Cp964) IBM-954C (Cp954c) IBM-954 (ibm954, Cp954) IBM-951 (Cp951, ibm951) IBM-949C (Cp949C, ibm949C) IBM-947 (Cp947, ibm947) IBM-943C (ibm943C, Cp943C) IBM-942C (Cp942C, ibm942C) IBM-939 (Cp5035, 5035) IBM-932 (ibm932, Cp932) IBM-930 (Cp5026, 5026) IBM-927 (ibm927, Cp927) IBM-924 (Cp924, ibm924) IBM-918 (ibm918, Cp918) IBM-897 (Cp897, ibm897) IBM-867 (Cp867, ibm867) IBM-1380 (Cp1380, ibm1380) IBM-1371 (Cp1371, ibm1371) IBM-1370 (Cp1370, ibm1370) IBM-1364 (Cp1364, ibm1364) IBM-1363C (ibm1363C, Cp1363C) IBM-1047 (Cp1047, ibm1047) IBM-1088 (Cp1088, ibm1088) IBM-1382 (ibm1382, Cp1382) IBM-1385 (Cp1385, ibm1385) IBM-1386 (ibm1386, Cp1386) IBM-1388 (Cp1388, ibm1388) IBM-1390 (Cp1390, ibm1390) IBM-1399 (ibm1399, Cp1399) IBM-290 (ibm290, Cp290) IBM-300 (Cp300, ibm300) IBM-301 (Cp301, ibm301) IBM-33722 (5050, Cp5050) X-UnicodeLittle (UnicodeLittle) X-UnicodeBig (UnicodeBig) IBM-864S (ibm864S, Cp864S) IBM-859 (Cp859, ibm859) IBM-858 (Cp858, ibm858) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
AIX	<p> IBM-837 (ibm837, Cp837) IBM-836 (ibm836, Cp836) IBM-835 (ibm835, Cp835) IBM-833 (ibm833, Cp833) IBM-808 (Cp808, ibm808) IBM-720 (Cp720, ibm720) IBM-420S (Cp420S, ibm420S) IBM-33722C (ibm-eucjp, Cp33722c) IBM-1046S (ibm1046S, Cp1046S) IBM-1043 (Cp1043, ibm1043) IBM-1041 (Cp1041, ibm1041) IBM-1027 (Cp1027, ibm1027) CESU-8 (CESU8) COMPOUND_TEXT (x-texte-composé, x11-compound-text) GB2312 (gb2312-1980, gb2312-80) GBK (GBK) hp-roman8 (roman8, ibm-1051, r8, Cp1051) IBM-1114 (Cp1114, ibm1114) IBM-1115 (Cp1115, ibm1115) IBM-1140 (ibm1140, Cp1140) IBM-1141 (Cp1141, ibm1141) IBM-1142 (Cp1142, ibm1142) IBM-1143 (Cp1143, ibm1143) IBM-1144 (ibm1144, Cp1144) IBM-1145 (Cp1145, ibm1145) IBM-1146 (Cp1146, ibm1146) IBM-1147 (Cp1147, ibm1147) IBM-1148 (ibm1148, Cp1148) IBM-1149 (Cp1149, ibm1149) IBM-1351 (Cp1351, ibm1351) IBM-1362 (Cp1362, ibm1362) IBM-1363 (ibm1363, Cp1363) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
Windows	<p> windows-1256S (Cp1256s, ibm-1256s) UTF-8J (UTF8J) UTF-32LE (UTF_32LE, X-UTF-32LE, UTF32LE) UTF-32BE (UTF_32BE, X-UTF-32BE, UTF32BE) PTCP154 (PT154, IBM-1169, Cyrillique-Asiatique, csPTCP154) KOI8-RU (ibm-1168, koi8_ru) ISO-8859-16 (8859-16, iso8859_16, iso8859-16) ISO-8859-14 (ISO_8859-14:1998, 8859-14, latin8, iso-ir-199, iso8859-14, l8, isoceltique, iso8859_14) IBM01141 (cp1141, ccsid01141, cp01141, 1141) IBM01142 (cp01142, cp1142, 1142, ccsid01142) IBM01143 (cp01143, 1143, ccsid01143, cp1143) IBM01144 (cp01144, cp1144, ccsid01144, 1144) IBM01145 (cp1145, cp01145, ccsid01145, 1145) IBM01146 (ccsid01146, cp01146, cp1146, 1146) IBM01147 (ccsid01147, cp1147, 1147, cp01147) IBM01148 (cp1148, ccsid01148, 1148, cp01148) IBM01149 (cp1149, cp01149, ccsid01149, 1149) IBM1047 (cp1047, 1047, ibm-1047) ISO-2022-JP-2 (csISO2022JP2, iso2022jp2) x-Big5-Solaris (Big5_Solaris) x-eucJP-Open (EUC_JP_Solaris, eucJP-open) x-IBM33722 (ibm33722, 33722, ibm-33722_vascii_vpua, ibm-5050, ibm-33722, cp33722) x-IBM930 (cp930, ibm930, ibm-930, 930) x-IBM939 (ibm-939, ibm939, cp939, 939) x-IBM964 (964, cp964, ibm-964, ibm964) x-ISO-2022-CN-CNS (ISO-2022-CN-CNS, ISO2022CN_CNS) x-iso-8859-11 (iso-8859-11, iso8859_11) x-JISAutoDetect (JISAutoDetect) x-MS932_0213 () x-MS950-HKSCS (MS950_HKSCS) x-PCK (pck) x-IBM1363C (ibm1363c, cp1363c, ibm-1363c) x-IBM420S (420s, ibm-420s, csibm420s, ibm420s, cp420s) x-IBM864S (csibm864s, ibm864s, cp864s, 864s, ibm-864s) x-IBM943C (cp943c, 943c, ibm-943c, ibm943c) x-IBM949C (ibm949c, cp949c, 949c, ibm-949c) x-IBM954C (cp954c, 954c, ibm-954c, ibm954c) x-ISO-8859-6S (8859_6s, iso8859-6s, iso8859_6s, iso-8859-6s) x-JIS0208 (jis_c6226-1983, jis_x0208-1983, csiso87jisx0208, x0208, iso-ir-87, jis0208) x-KSC5601 (ksc5601) x-MacDingbat (macdingbat) x-MacHebrew (machebrew) x-MacThai (macthai) x-MacUkraine (macukraine) x-IBM1046S (ibm-1046s, 1046s, cp1046s, ibm1046s) x-IBM-udcJP (IBM-udcJP) JIS_X0201 (jis_x0201, x0201, cshalfwidthkatakana, jis0201) IBM-939A (Cp939A, ibm939A) IBM-930A (ibm930A, Cp930A) IBM-33722A (Cp33722A, ibm33722A) x-windows-iso2022jp (windows-iso2022jp) x-windows-50221 (ms50221, cp50221) x-windows-50220 (cp50220, ms50220) X-UTF-32LE-BOM (UTF_32LE_BOM, UTF-32LE-BOM) X-UTF-32BE-BOM (UTF_32BE_BOM, UTF-32BE-BOM) x-SJIS_0213 () IBM01140 (ccsid01140, cp01140, 1140, cp1140) IBM00858 (cp858, ccsid00858, 858, cp00858) X-UnicodeLittle (UnicodeLittle) X-UnicodeBig (UnicodeBig) IBM-859 (Cp859, ibm859) IBM-837 (ibm837, Cp837) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
Windows	<p> IBM-836 (ibm836, Cp836) IBM-835 (ibm835, Cp835) IBM-833 (ibm833, Cp833) IBM-808 (Cp808, ibm808) IBM-720 (Cp720, ibm720) IBM-33722C (ibm-eucjp, Cp33722c) IBM-301 (Cp301, ibm301) IBM-300 (Cp300, ibm300) IBM-290 (ibm290, Cp290) IBM-1399 (ibm1399, Cp1399) IBM-1390 (Cp1390, ibm1390) IBM-1388 (Cp1388, ibm1388) IBM-1385 (Cp1385, ibm1385) IBM-1382 (ibm1382, Cp1382) IBM-1088 (Cp1088, ibm1088) IBM-1043 (Cp1043, ibm1043) IBM-1041 (Cp1041, ibm1041) IBM-1027 (Cp1027, ibm1027) CESU-8 (CESU8) COMPOUND_TEXT (x-texte-composé, x11-compound-text) GB2312 (gb2312-1980, gb2312-80) GBK (GBK) hp-roman8 (roman8, ibm-1051, r8, Cp1051) IBM-1115 (Cp1115, ibm1115) IBM-1351 (Cp1351, ibm1351) IBM-1362 (Cp1362, ibm1362) IBM-1363 (ibm1363, Cp1363) IBM-1364 (Cp1364, ibm1364) IBM-1370 (Cp1370, ibm1370) IBM-1371 (Cp1371, ibm1371) IBM-1380 (Cp1380, ibm1380) IBM-867 (Cp867, ibm867) IBM-897 (Cp897, ibm897) IBM-924 (Cp924, ibm924) IBM-927 (ibm927, Cp927) IBM-932 (ibm932, Cp932) IBM-947 (Cp947, ibm947) IBM-951 (Cp951, ibm951) IBM-954 (ibm954, Cp954) IBM-971 (Cp971, ibm971) ISO-8859-10 (latin6, 8859-10, ISO_8859-10:1992, iso8859_10, iso-ir-157, ibm-919, iso8859-10, l6, csisolatin6) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
Red Hat Enterprise Linux sur System x	<p> windows-1256S (Cp1256s, ibm-1256s) UTF-8J (UTF8J) UTF-32LE (UTF_32LE, X-UTF-32LE, UTF32LE) UTF-32BE (UTF_32BE, X-UTF-32BE, UTF32BE) UTF-32 (UCS-4, UTF32, ISO-10646-UCS-4) PTCP154 (PT154, IBM-1169, Cyrillique-Asiatique, csPTCP154) KOI8-RU (ibm-1168, koi8_ru) ISO-8859-16 (8859-16, iso8859_16, iso8859-16) ISO-8859-14 (ISO_8859-14:1998, 8859-14, latin8, iso-ir-199, iso8859-14, l8, isoceltique, iso8859_14) IBM01141 (cp1141, ccsid01141, cp01141, 1141) IBM01142 (cp01142, cp1142, 1142, ccsid01142) IBM01143 (cp01143, 1143, ccsid01143, cp1143) IBM01144 (cp01144, cp1144, ccsid01144, 1144) IBM01145 (cp1145, cp01145, ccsid01145, 1145) IBM01146 (ccsid01146, cp01146, cp1146, 1146) IBM01147 (ccsid01147, cp1147, 1147, cp01147) IBM01148 (cp1148, ccsid01148, 1148, cp01148) IBM01149 (cp1149, cp01149, ccsid01149, 1149) IBM1047 (cp1047, 1047, ibm-1047) IBM918 (cp918, ebcdic-cp-ar2, ibm-918, 918) ISO-2022-JP-2 (csISO2022JP2, iso2022jp2) x-Big5-Solaris (Big5_Solaris) x-eucJP-Open (EUC_JP_Solaris, eucJP-open) x-IBM33722 (ibm33722, 33722, ibm-33722_vascii_vpua, ibm-5050, ibm-33722, cp33722) x-IBM930 (cp930, ibm930, ibm-930, 930) x-IBM939 (ibm-939, ibm939, cp939, 939) x-IBM964 (964, cp964, ibm-964, ibm964) x-ISO-2022-CN-CNS (ISO-2022-CN-CNS, ISO2022CN_CNS) x-iso-8859-11 (iso-8859-11, iso8859_11) x-JISAutoDetect (JISAutoDetect) x-MS932_0213 () x-MS950-HKSCS (MS950_HKSCS) x-PCK (pck) x-IBM1363C (ibm1363c, cp1363c, ibm-1363c) x-IBM420S (420s, ibm-420s, csibm420s, ibm420s, cp420s) x-IBM864S (csibm864s, ibm864s, cp864s, 864s, ibm-864s) x-IBM943C (cp943c, 943c, ibm-943c, ibm943c) x-IBM949C (ibm949c, cp949c, 949c, ibm-949c) x-IBM954C (cp954c, 954c, ibm-954c, ibm954c) x-ISO-8859-6S (8859_6s, iso8859-6s, iso8859_6s, iso-8859-6s) x-JIS0208 (jis_c6226-1983, jis_x0208-1983, csiso87jisx0208, x0208, iso-ir-87, jis0208) x-KSC5601 (ksc5601) x-MacDingbat (macdingbat) x-MacHebrew (machebrew) x-MacThai (macthai) x-MacUkraine (macukraine) x-IBM1046S (ibm-1046s, 1046s, cp1046s, ibm1046s) x-IBM-udcJP (IBM-udcJP) JIS_X0201 (jis_x0201, x0201, cshalfwidthkatakana, jis0201) IBM-939A (Cp939A, ibm939A) IBM-930A (ibm930A, Cp930A) IBM-33722A (Cp33722A, ibm33722A) x-windows-iso2022jp (windows-iso2022jp) x-windows-50221 (ms50221, cp50221) x-windows-50220 (cp50220, ms50220) X-UTF-32LE-BOM (UTF_32LE_BOM, UTF-32LE-BOM) X-UTF-32BE-BOM (UTF_32BE_BOM, UTF-32BE-BOM) x-SJIS_0213 () IBM01140 (ccsid01140, cp01140, 1140, cp1140) IBM00858 (cp858, ccsid00858, 858, cp00858) X-UnicodeLittle (UnicodeLittle) X-UnicodeBig (UnicodeBig) IBM-859 (Cp859, ibm859) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
Red Hat Enterprise Linux sur System x	<p> IBM-837 (ibm837, Cp837) IBM-836 (ibm836, Cp836) IBM-835 (ibm835, Cp835) IBM-833 (ibm833, Cp833) IBM-808 (Cp808, ibm808) IBM-720 (Cp720, ibm720) IBM-33722C (ibm-eucjp, Cp33722c) IBM-301 (Cp301, ibm301) IBM-300 (Cp300, ibm300) IBM-290 (ibm290, Cp290) IBM-1399 (ibm1399, Cp1399) IBM-1390 (Cp1390, ibm1390) IBM-1388 (Cp1388, ibm1388) IBM-1385 (Cp1385, ibm1385) IBM-1382 (ibm1382, Cp1382) IBM-1088 (Cp1088, ibm1088) IBM-1043 (Cp1043, ibm1043) IBM-1041 (Cp1041, ibm1041) IBM-1027 (Cp1027, ibm1027) CESU-8 (CESU8) COMPOUND_TEXT (x-texte-composé, x11-compound-text) GB2312 (gb2312-1980, gb2312-80) GBK (GBK) hp-roman8 (roman8, ibm-1051, r8, Cp1051) IBM-1114 (Cp1114, ibm1114) IBM-1115 (Cp1115, ibm1115) IBM-1351 (Cp1351, ibm1351) IBM-1362 (Cp1362, ibm1362) IBM-1363 (ibm1363, Cp1363) IBM-1364 (Cp1364, ibm1364) IBM-1370 (Cp1370, ibm1370) IBM-1371 (Cp1371, ibm1371) IBM-1380 (Cp1380, ibm1380) IBM-867 (Cp867, ibm867) IBM-897 (Cp897, ibm897) IBM-924 (Cp924, ibm924) IBM-927 (ibm927, Cp927) IBM-932 (ibm932, Cp932) IBM-947 (Cp947, ibm947) IBM-951 (Cp951, ibm951) IBM-954 (ibm954, Cp954) IBM-971 (Cp971, ibm971) ISO-8859-10 (latin6, 8859-10, ISO_8859-10:1992, iso8859_10, iso-ir-157, ibm-919, iso8859-10, l6, csisolatin6) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
<p>z/OS z/OS z/OS</p>	<p>windows-1256S (Cp1256s, ibm-1256s) UTF-8J (UTF8J) UTF-32LE (UTF_32LE, X-UTF-32LE, UTF32LE) UTF-32BE (UTF_32BE, X-UTF-32BE, UTF32BE) UTF-32 (UCS-4, UTF32, ISO-10646-UCS-4) UTF-16 (UTF16, Unicode, UTF_16, UCS-2) Shift_JIS () PTCP154 (PT154, IBM-1169, Cyrillique-Asiatique, csPTCP154) MacUkraine () MacThai () MacHebrew () MacDingbat () KSC5601 () KOI8-U (koi8_u, ibm-1167) KOI8-RU (ibm-1168, koi8_ru) Johab (x-johab) JISO208 () JISO201 () ISO-8859-6S (iso8859-6S, iso8859_6S) ISO-8859-16 (8859-16, iso8859_16, iso8859-16) ISO-8859-14 (ISO_8859-14:1998, 8859-14, latin8, iso-ir-199, iso8859-14, l8, isoceltic, iso8859_14) ISO-8859-10 (latin6, 8859-10, ISO_8859-10:1992, iso8859_10, iso-ir-157, ibm-919, iso8859-10, l6, csisolatin6) IBM-971 (Cp971, ibm971) IBM-964 (ibm-euctw, Cp964) IBM-954C (Cp954c) IBM-954 (ibm954, Cp954) IBM-951 (Cp951, ibm951) IBM-949C (Cp949C, ibm949C) IBM-947 (Cp947, ibm947) IBM-943C (ibm943C, Cp943C) IBM-942C (Cp942C, ibm942C) IBM-939 (Cp5035, 5035) IBM-932 (ibm932, Cp932) IBM-930 (Cp5026, 5026) IBM-927 (ibm927, Cp927) IBM-924 (Cp924, ibm924) IBM-918 (ibm918, Cp918) IBM-897 (Cp897, ibm897) IBM-867 (Cp867, ibm867) IBM-1380 (Cp1380, ibm1380) IBM-1371 (Cp1371, ibm1371) IBM-1370 (Cp1370, ibm1370) IBM-1364 (Cp1364, ibm1364) IBM-1363C (ibm1363C, Cp1363C) IBM-1363 (ibm1363, Cp1363) IBM-1088 (Cp1088, ibm1088) IBM-1382 (ibm1382, Cp1382) IBM-1385 (Cp1385, ibm1385) IBM-1386 (ibm1386, Cp1386) IBM-1388 (Cp1388, ibm1388) IBM-1390 (Cp1390, ibm1390) IBM-1399 (ibm1399, Cp1399) IBM-290 (ibm290, Cp290) IBM-300 (Cp300, ibm300) IBM-301 (Cp301, ibm301) IBM-33722 (5050, Cp5050) IBM-33722C (ibm-eucjp, Cp33722c) IBM-930A (ibm930A, Cp930A) X-UnicodeLittle (UnicodeLittle) X-UnicodeBig (UnicodeBig) IBM-864S (ibm864S, Cp864S) IBM-859 (Cp859, ibm859) IBM-858 (Cp858, ibm858)</p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
  z/OS	<p> IBM-837 (ibm837, Cp837) IBM-836 (ibm836, Cp836) IBM-835 (ibm835, Cp835) IBM-833 (ibm833, Cp833) IBM-808 (Cp808, ibm808) IBM-720 (Cp720, ibm720) IBM-420S (Cp420S, ibm420S) IBM-1047 (Cp1047, ibm1047) IBM-1046S (ibm1046S, Cp1046S) IBM-1043 (Cp1043, ibm1043) IBM-1041 (Cp1041, ibm1041) IBM-1027 (Cp1027, ibm1027) CESU-8 (CESU8) COMPOUND_TEXT (x-texte-composé, x11-compound-text) GB2312 (gb2312-1980, gb2312-80) GBK (GBK) hp-roman8 (roman8, ibm-1051, r8, Cp1051) IBM-1114 (Cp1114, ibm1114) IBM-1115 (Cp1115, ibm1115) IBM-1140 (ibm1140, Cp1140) IBM-1141 (Cp1141, ibm1141) IBM-1142 (Cp1142, ibm1142) IBM-1143 (Cp1143, ibm1143) IBM-1144 (ibm1144, Cp1144) IBM-1145 (Cp1145, ibm1145) IBM-1146 (Cp1146, ibm1146) IBM-1147 (Cp1147, ibm1147) IBM-1148 (ibm1148, Cp1148) IBM-1149 (Cp1149, ibm1149) IBM-1351 (Cp1351, ibm1351) IBM-1362 (Cp1362, ibm1362) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
Linux on POWER Systems - Big Endian	<p> windows-1256S (Cp1256s, ibm-1256s) UTF-8J (UTF8J) UTF-32LE (UTF_32LE, X-UTF-32LE, UTF32LE) UTF-32BE (UTF_32BE, X-UTF-32BE, UTF32BE) UTF-32 (UCS-4, UTF32, ISO-10646-UCS-4) UTF-16 (UTF16, Unicode, UTF_16, UCS-2) Shift_JIS () PTCP154 (PT154, IBM-1169, Cyrillique-Asiatique, csPTCP154) MacUkraine () MacThai () MacHebrew () MacDingbat () KSC5601 () KOI8-U (koi8_u, ibm-1167) KOI8-RU (ibm-1168, koi8_ru) Johab (x-johab) JISO208 () JISO201 () ISO-8859-6S (iso8859-6S, iso8859_6S) ISO-8859-16 (8859-16, iso8859_16, iso8859-16) ISO-8859-14 (ISO_8859-14:1998, 8859-14, latin8, iso-ir-199, iso8859-14, l8, isoceltic, iso8859_14) ISO-8859-10 (latin6, 8859-10, ISO_8859-10:1992, iso8859_10, iso-ir-157, ibm-919, iso8859-10, l6, csisolatin6) IBM-971 (Cp971, ibm971) IBM-964 (ibm-euctw, Cp964) IBM-954C (Cp954c) IBM-954 (ibm954, Cp954) IBM-951 (Cp951, ibm951) IBM-949C (Cp949C, ibm949C) IBM-947 (Cp947, ibm947) IBM-943C (ibm943C, Cp943C) IBM-942C (Cp942C, ibm942C) IBM-939 (Cp5035, 5035) IBM-932 (ibm932, Cp932) IBM-930 (Cp5026, 5026) IBM-927 (ibm927, Cp927) IBM-924 (Cp924, ibm924) IBM-918 (ibm918, Cp918) IBM-897 (Cp897, ibm897) IBM-867 (Cp867, ibm867) IBM-1380 (Cp1380, ibm1380) IBM-1371 (Cp1371, ibm1371) IBM-1370 (Cp1370, ibm1370) IBM-1364 (Cp1364, ibm1364) IBM-1363C (ibm1363C, Cp1363C) IBM-1047 (Cp1047, ibm1047) IBM-1088 (Cp1088, ibm1088) IBM-1382 (ibm1382, Cp1382) IBM-1385 (Cp1385, ibm1385) IBM-1386 (ibm1386, Cp1386) IBM-1388 (Cp1388, ibm1388) IBM-1390 (Cp1390, ibm1390) IBM-1399 (ibm1399, Cp1399) IBM-290 (ibm290, Cp290) IBM-300 (Cp300, ibm300) IBM-301 (Cp301, ibm301) IBM-33722 (5050, Cp5050) X-UnicodeLittle (UnicodeLittle) X-UnicodeBig (UnicodeBig) IBM-864S (ibm864S, Cp864S) IBM-859 (Cp859, ibm859) IBM-858 (Cp858, ibm858) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
Linux on POWER Systems - Big Endian	<p> IBM-837 (ibm837, Cp837) IBM-836 (ibm836, Cp836) IBM-835 (ibm835, Cp835) IBM-833 (ibm833, Cp833) IBM-808 (Cp808, ibm808) IBM-720 (Cp720, ibm720) IBM-420S (Cp420S, ibm420S) IBM-33722C (ibm-eucjp, Cp33722c) IBM-1046S (ibm1046S, Cp1046S) IBM-1043 (Cp1043, ibm1043) IBM-1041 (Cp1041, ibm1041) IBM-1027 (Cp1027, ibm1027) CESU-8 (CESU8) COMPOUND_TEXT (x-texte-composé, x11-compound-text) GB2312 (gb2312-1980, gb2312-80) GBK (GBK) hp-roman8 (roman8, ibm-1051, r8, Cp1051) IBM-1114 (Cp1114, ibm1114) IBM-1115 (Cp1115, ibm1115) IBM-1140 (ibm1140, Cp1140) IBM-1141 (Cp1141, ibm1141) IBM-1142 (Cp1142, ibm1142) IBM-1143 (Cp1143, ibm1143) IBM-1144 (ibm1144, Cp1144) IBM-1145 (Cp1145, ibm1145) IBM-1146 (Cp1146, ibm1146) IBM-1147 (Cp1147, ibm1147) IBM-1148 (ibm1148, Cp1148) IBM-1149 (Cp1149, ibm1149) IBM-1351 (Cp1351, ibm1351) IBM-1362 (Cp1362, ibm1362) IBM-1363 (ibm1363, Cp1363) </p>

Plateforme	Codages pris en charge (pas dans la table des codages communs)
HP (PA-RISC)	<p> UTF-32LE (UTF_32LE, X-UTF-32LE, UTF32LE) UTF-32BE (UTF_32BE, X-UTF-32BE, UTF32BE) IBM01147 (ccsid01147, cp1147, 1147, cp01147) IBM01148 (cp1148, ccsid01148, 1148, cp01148) IBM01149 (cp1149, cp01149, ccsid01149, 1149) IBM1047 (cp1047, 1047, ibm-1047) IBM918 (cp918, ebcdic-cp-ar2, ibm-918, 918) ISO-2022-JP-2 (csISO2022JP2, iso2022jp2) Roman9 (Roman9) x-Big5-Solaris (Big5_Solaris) x-eucJP-Open (EUC_JP_Solaris, eucJP-open) x-IBM33722 (ibm33722, 33722, ibm-33722_vascii_vpua, ibm-5050, ibm-33722, cp33722) x-IBM930 (cp930, ibm930, ibm-930, 930) x-IBM939 (ibm-939, ibm939, cp939, 939) x-windows-iso2022jp (windows-iso2022jp) x-windows-50221 (ms50221, cp50221) x-windows-50220 (cp50220, ms50220) X-UTF-32LE-BOM (UTF_32LE_BOM, UTF-32LE-BOM) X-UTF-32BE-BOM (UTF_32BE_BOM, UTF-32BE-BOM) x-SJIS_0213 () x-PCK (pck) x-MS950-HKSCS (MS950_HKSCS) x-MS932_0213 () x-JISAutoDetect (JISAutoDetect) x-iso-8859-11 (iso-8859-11, iso8859_11) x-ISO-2022-CN-CNS (ISO-2022-CN-CNS, ISO2022CN_CNS) x-IBM964 (964, cp964, ibm-964, ibm964) IBM01146 (ccsid01146, cp01146, cp1146, 1146) IBM01145 (cp1145, cp01145, ccsid01145, 1145) IBM01144 (cp01144, cp1144, ccsid01144, 1144) IBM01143 (cp01143, 1143, ccsid01143, cp1143) IBM01142 (cp01142, cp1142, 1142, ccsid01142) IBM01141 (cp1141, ccsid01141, cp01141, 1141) IBM01140 (ccsid01140, cp01140, 1140, cp1140) IBM00858 (cp858, ccsid00858, 858, cp00858) X-UnicodeLittle (UnicodeLittle) X-UnicodeBig (UnicodeBig) COMPOUND_TEXT (x-texte-composé, x11-compound-text) hp-roman8 (roman8, ibm-1051, r8, Cp1051) IBM-1364 (Cp1364, ibm1364) IBM-942C (Cp942C, ibm942C) IBM-943C (ibm943C, Cp943C) IBM-949C (Cp949C, ibm949C) JIS0201 () JIS0208 () KOI8-U (koi8_u, ibm-1167) MacDingbat () MacHebrew () MacThai () MacUkraine () UTF-32 (UCS-4, UTF32, ISO-10646-UCS-4) </p>

Plateformes par codage

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
x-MacUkraine	macukraine	Solaris, SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
x-MacThai	macthai	Solaris, SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
x-MacHebrew	MacHebrew	Solaris, SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
x-MacDingbat	mactdingbat	Solaris, SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
x-KSC5601	ksc5601	Solaris, SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
x-JIS0208	jis_c6226-1983, jis_x0208-1983, csiso87jisx0208, x0208, iso-ir-87, jis0208	Solaris, SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
x-ISO-8859-6S	8859_6s, iso8859-6s, iso8859_6s, iso-8859-6s	SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
x-IBM954C	cp954c, 954c, ibm-954c, ibm954c	SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
x-IBM949C	ibm949c, cp949c, 949c, ibm-949c	Solaris, SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
x-IBM943C	cp943c, 943c, ibm-943c, ibm943c	Solaris, SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
x-IBM864S	csibm864s, ibm864s, cp864s, 864s, ibm-864s	SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
x-IBM420S	420s, ibm-420s, csibm420s, ibm420s, cp420s	SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
x-IBM1363C	ibm1363c, cp1363c, ibm-1363c	SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
x-IBM1046S	ibm-1046s, 1046s, cp1046s, ibm1046s	SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
x-IBM-udcJP	IBM-udcJP	SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
JIS_X0201	jis_x0201, x0201, cshalfwidthkatakana, jis0201	Solaris, SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
IBM-939A	Cp939A, ibm939A	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
IBM-930A	ibm930A, Cp930A	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x,  z/OS

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
  IBM-924_LF	Cp924_LF, ibm924_LF	IBM i
IBM-33722A	Cp33722A, ibm33722A	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z Windows, Red Hat Enterprise Linux sur System x
  IBM-1141_LF	Cp1141_LF, ibm1141_LF	IBM i
  IBM-1047_LF	Cp1047_LF, ibm1047_LF	IBM i
x-windows-iso2022jp	windows-iso2022jp	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
x-windows-50221	ms50221, cp50221	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
x-windows-50220	cp50220, ms50220	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
X-UTF-32LE-BOM	UTF_32LE_BOM, UTF-32LE-BOM	Solaris, SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
X-UTF-32BE-BOM	UTF_32BE_BOM, UTF-32BE-BOM	Solaris, SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
x-SJIS_0213		Solaris, SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
x-PCK	pck	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
x-MS950-HKSCS	MS950_HKSCS	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
x-MS932_0213		Solaris, SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
x-JISAutoDetect	JISAutoDetect	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
x-iso-8859-11	iso-8859-11, iso8859_11	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
x-ISO-2022-CN-CNS	ISO-2022-CN-CNS, ISO2022CN_CNS	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
x-IBM964	964, cp964, ibm-964, ibm964	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
x-IBM939	ibm-939, ibm939, cp939, 939	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
x-IBM930	cp930, ibm930, ibm-930, 930	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
x-IBM33722	ibm33722, 33722, ibm-33722_vascii_vpua, ibm-5050, ibm-33722, cp33722	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
x-eucJP-Ouvert	EUC_JP_Solaris, eucJP-open	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
x-Big5-Solaris	Big5_Solaris	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
Roman9	Roman9	HP (PA-RISC)
ISO-2022-JP-2	csISO2022JP2, iso2022jp2	Solaris, SUSE Linux Enterprise Server sous System x, Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
IBM918	cp918, ebcdic-cp-ar2, ibm-918, 918	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
IBM1047	cp1047, 1047, ibm-1047	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
IBM01149	cp1149, cp01149, ccsid01149, 1149	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
IBM01148	cp1148, ccsid01148, 1148, cp01148	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
IBM01147	ccsid01147, cp1147, 1147, cp01147	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
IBM01146	ccsid01146, cp01146, cp1146, 1146	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
IBM01145	cp1145, cp01145, ccsid01145, 1145	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
IBM01144	cp01144, cp1144, ccsid01144, 1144	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
IBM01143	cp01143, 1143, ccsid01143, cp1143	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
IBM01142	cp01142, cp1142, 1142, ccsid01142	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
IBM01141	cp1141, ccsid01141, cp01141, 1141	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
IBM01140	ccsid01140, cp01140, 1140, cp1140	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
IBM00858	cp858, ccsid00858, 858, cp00858	Solaris, SUSE Linux Enterprise Server sous System x, HP-UX (Itanium), Linux pour IBM Z Windows, Red Hat Enterprise Linux sous System x, HP (PA-RISC)
X-UnicodeLittle	UnicodeLittle	Solaris, SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z  AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
X-UnicodeBig	UnicodeBig	Solaris, SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)
IBM-864S	ibm864S, Cp864S	 IBM i,  AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-859	Cp859, ibm859	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-858	Cp858, ibm858	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-837	ibm837, Cp837	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
IBM-836	ibm836, Cp836	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-835	ibm835, Cp835	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-833	ibm833, Cp833	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-808	Cp808, ibm808	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
IBM-720	Cp720, ibm720	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-420S	Cp420S, ibm420S	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-33722C	ibm-eucjp, Cp33722c	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-33722	5050, Cp5050	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-301	Cp301, ibm301	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
IBM-300	Cp300, ibm300	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-290	ibm290, Cp290	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-1399	ibm1399, Cp1399	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-1390	Cp1390, ibm1390	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
IBM-1388	Cp1388, ibm1388	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-1386	ibm1386, Cp1386	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-1385	Cp1385, ibm1385	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-1382	ibm1382, Cp1382	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-1088	Cp1088, ibm1088	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
IBM-1047	Cp1047, ibm1047	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-1046S	ibm1046S, Cp1046S	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-1043	Cp1043, ibm1043	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-1041	Cp1041, ibm1041	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-1027	Cp1027, ibm1027	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
CESU-8	CESU8	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
COMPOUND_TEXTE	x-composé-texte, x11-compound-text	Solaris, SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)
GB2312	gb2312-1980, gb2312-80	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
GBK	GBK	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
hp-roman8	roman8, ibm-1051, r8, Cp1051	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)
IBM-1114	Cp1114, ibm1114	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-1115	Cp1115, ibm1115	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-1140	ibm1140, Cp1140	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-1141	Cp1141, ibm1141	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-1142	Cp1142, ibm1142	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
IBM-1143	Cp1143, ibm1143	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-1144	ibm1144, Cp1144	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-1145	Cp1145, ibm1145	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-1146	Cp1146, ibm1146	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-1147	Cp1147, ibm1147	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-1148	ibm1148, Cp1148	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-1149	Cp1149, ibm1149	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
IBM-1351	Cp1351, ibm1351	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-1362	Cp1362, ibm1362	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-1363	ibm1363, Cp1363	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-1363C	ibm1363C, Cp1363C	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-1364	Cp1364, ibm1364	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
IBM-1370	Cp1370, ibm1370	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-1371	Cp1371, ibm1371	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-1380	Cp1380, ibm1380	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-867	Cp867, ibm867	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
IBM-897	Cp897, ibm897	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-918	ibm918, Cp918	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-924	Cp924, ibm924	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-927	ibm927, Cp927	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-930	Cp5026, 5026	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
IBM-932	ibm932, Cp932	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-939	Cp5035, 5035	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-942C	Cp942C, ibm942C	Solaris,  IBM i, HP-UX (Itanium), AIX,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)
IBM-943C	ibm943C, Cp943C	 IBM i, HP-UX (Itanium), AIX,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)
IBM-947	Cp947, ibm947	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
IBM-949C	Cp949C, ibm949C	 IBM i, HP-UX (Itanium), AIX,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)
IBM-951	Cp951, ibm951	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-954	ibm954, Cp954	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
IBM-954C	Cp954c	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-964	ibm-euctw, Cp964	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
IBM-971	Cp971, ibm971	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
ISO-8859-10	latin6, 8859-10, ISO_8859-10:1992, iso8859_10, iso-ir-157, ibm-919, iso8859-10, l6, csisolatin6	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
ISO-8859-14	ISO_8859-14:1998, 8859-14, latin8, iso-ir-199, iso8859-14, l8, isoceltique, iso8859_14	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
ISO-8859-16	8859-16, iso8859_16, iso8859-16	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
ISO-8859-6S	iso8859-6S, iso8859_6S	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
JIS0201		 IBM i, HP-UX (Itanium), AIX,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
JIS0208		 IBM i, HP-UX (Itanium), AIX,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)
JOHAB	x-johab	 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
KO18-RU	ibm-1168, koi8_ru	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
KO18-U	koi8_u, ibm-1167	Solaris,  IBM i, AIX,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)
KSC5601		 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
MacDingbat		 IBM i, HP-UX (Itanium), AIX,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
MacHebrew		 IBM i, HP-UX (Itanium), AIX,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)
MacThai		 IBM i, HP-UX (Itanium), AIX,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)
MacUkraine		 IBM i, HP-UX (Itanium), AIX,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)
PTCP154	PT154, IBM-1169, Cyrillique-Asiatique, csPTCP154	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian
Shift_JIS		 IBM i, AIX,  z/OS, Linux on POWER Systems - Big Endian
UTF-16	UTF16, Unicode, UTF_16, UCS-2	 IBM i, HP-UX (Itanium), AIX,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
UTF-32	UCS-4, UTF32, ISO-10646-UCS-4	Solaris, SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)
UTF-32BE	UTF_32BE, X-UTF-32BE, UTF32BE	Solaris, SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)
UTF-32LE	UTF_32LE, X-UTF-32LE, UTF32LE	Solaris, SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems-Big Endian, HP (PA-RISC)
UTF-8J	UTF8J	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian

Codage	Alias	Plateformes sur lesquelles ce codage est pris en charge
windows-1256S	Cp1256s, ibm-1256s	SUSE Linux Enterprise Server sous System x,  IBM i, Linux pour IBM Z AIX, Windows, Red Hat Enterprise Linux sous System x,  z/OS, Linux on POWER Systems - Big Endian

Concepts associés

«Utilisation de fichiers de définition de transfert», à la page 164

Vous pouvez spécifier un fichier de définition de transfert qui peut être utilisé pour créer un transfert de fichier. Le fichier de définition de transfert est un fichier XML qui définit tout ou partie des informations requises pour créer le transfert.

Référence associée

«Transfert de fichiers texte avec MFT», à la page 662

Le transfert de fichiers texte implique la conversion de la page de codes d'un fichier d'une page de codes à une autre. Le transfert de fichier texte implique également la conversion de caractères CRLF (retour chariot-saut de ligne) entre les systèmes. Cette rubrique récapitule le comportement de transfert de fichier texte de Managed File Transfer.

«fteCreateTransfer: démarrer un nouveau transfert de fichier», à la page 447

La commande **fteCreateTransfer** crée et démarre un nouveau transfert de fichier à partir de la ligne de commande. Cette commande peut démarrer un transfert de fichier immédiatement, planifier un transfert de fichier à une heure et une date ultérieures, répéter un transfert planifié une ou plusieurs fois et déclencher un transfert de fichiers sous certaines conditions.

Comment les agents MFT utilisent le segment de mémoire Java et le segment de mémoire natif

Un agent IBM MQ Managed File Transfer s'exécute en tant que processus Java . En tant que tel, l'agent s'exécute dans l'environnement virtualisé de la machine virtuelle Java (JVM).

La machine virtuelle Java elle-même est un processus natif, délimité par le matériel et le système d'exploitation. La machine virtuelle Java gère deux zones de mémoire:

- Le segment de mémoire Java

Il contient les instances des objets Java et est géré par le processus de récupération de place. La taille maximale du segment de mémoire Java est allouée lors du démarrage de la machine virtuelle Java à l'aide de l'option JVM **-Xmx** .

- Segment de mémoire natif

Le segment de mémoire natif contient des ressources pour la machine virtuelle Java elle-même ; par exemple, le compilateur Just-In-Time, les classes et les ClassLoaders.

Un agent utilise principalement le segment de mémoire Java . Lors de l'exécution de transferts gérés, l'agent utilise le segment de mémoire Java pour créer les objets Java requis pour le transfert. Toutes les données de fichier lues dans les mémoires tampon par l'agent sont également stockées dans le segment de mémoire Java .

Un agent ne contient pas lui-même de code qui utilise le segment de mémoire natif. Toutefois, il existe du code natif dans l'interface de mise en file d'attente de messages Java (JMQUI) que l'agent utilise pour communiquer avec son gestionnaire de files d'attente d'agent.

Ce code natif est utilisé lorsqu'un agent se connecte à son gestionnaire de files d'attente d'agent à l'aide du transport BINDINGS. Il s'agit d'une connexion de mémoire partagée locale (parfois appelée communication interprocessus ou IPC), plutôt que d'une connexion TCP/IP qui est utilisée si un agent se connecte à l'aide du transport CLIENT. Lorsqu'un agent est configuré pour utiliser le transport BINDINGS, le segment de mémoire natif est utilisé pour transmettre des messages et des commandes entre l'agent et le gestionnaire de files d'attente de l'agent.

Cela signifie qu'un agent fortement chargé qui est connecté à son gestionnaire de files d'attente d'agent à l'aide du transport BINDINGS utilise plus largement le segment de mémoire natif, par rapport à un agent équivalent qui est connecté à l'aide du transport CLIENT.

Une idée fausse courante est que le segment de mémoire Java d'un agent doit être égal (ou supérieur) à la taille du fichier le plus volumineux à transférer. Ceci n'est pas correct car les données de fichier sont lues en mémoire par étapes.

A titre indicatif, la quantité maximale de segments de mémoire Java utilisée pour stocker les données de fichier pour chaque transfert peut être calculée de la manière suivante:

```
Memory allocated for a transfer = agentCheckpointInterval *
agentFrameSize * agentWindowSize * agentChunkSize
```

Impact de l'utilisation du segment de mémoire Java et du segment de mémoire natif sur les agents

Lorsqu'un `java.lang.OutOfMemoryError` se produit, vous pouvez penser qu'il est raisonnable d'augmenter la quantité de segment de mémoire Java disponible pour l'application, à l'aide de la propriété système `-Xmx` Java. Par exemple, le paramètre de propriété suivant tente d'allouer une taille de segment de mémoire Java maximale de 2GB:

```
-Xmx2048M
```

Toutefois, l'allocation d'une trop grande quantité de segment de mémoire Java pour une application peut entraîner la création d'un `java.lang.OutOfMemoryError`, en raison de l'épuisement du segment de mémoire natif. En effet, à mesure que l'espace du segment de mémoire Java augmente, le segment de mémoire natif doit être réduit pour pouvoir s'y adapter.

Pour plus d'informations sur la manière d'empêcher les `java.lang.OutOfMemoryErrors` qui sont causées par l'épuisement du segment de mémoire natif, voir [Que faire si votre agent MFT ABENDS avec une erreur java.lang.OutOfMemoryError due à l'épuisement de la mémoire native.](#)

Formats de message XML utilisés par MFT

Managed File Transfer utilise des messages au format XML à diverses fins: pour commander un agent, pour consigner des informations sur les moniteurs, les planifications et les transferts et pour définir les informations utilisées pour la configuration. Structure logique des formats XML utilisés à ces fins et décrits par le schéma XML.

Chaque version de Managed File Transfer utilise un schéma XML pour valider les messages écrits en XML. L'agent extrait la version du schéma XML et détermine si le schéma est pris en charge.

Une fois que vous avez installé Managed File Transfer, vous pouvez trouver les fichiers de schéma de message Managed File Transfer dans le répertoire suivant: `MQ_INSTALLATION_PATH/mqft/samples/schema`. Les schémas suivants sont inclus:

Schémas des messages XML pouvant être placés dans une file d'attente de commandes d'agent

`FileTransfer.xsd`

`Internal.xsd`

`Monitor.xsd`

`PingAgent.xsd`

Pour plus d'informations sur l'insertion de messages XML dans une file d'attente de commandes d'agent, voir [«Contrôle de MFT via le placement de messages dans la file d'attente de commandes d'agent»](#), à la page 276.

Schémas des messages XML publiés dans SYSTEM.FTE

MonitorList.xsd

MonitorLog.xsd

ScheduleList.xsd

ScheduleLog.xsd

TransferLog.xsd

TransferStatus.xsd

Pour plus d'informations sur les messages XML publiés dans SYSTEM.FTE et la structure de SYSTEM.FTE, voir [«Rubrique SYSTEM.FTE»](#), à la page 636.

Autres schémas utilisés par Managed File Transfer

fteutils.xsd: Ce schéma contient des définitions d'élément communes et est inclus par certains des autres schémas.

Notification.xsd

ProtocolBridgeCredentials.xsd

ProtocolBridgeProperties.xsd

ConnectDirectCredentials.xsd

ConnectDirectNodeProperties.xsd

ConnectDirectProcessDefinitions.xsd

Reply.xsd

UserSandboxes.xsd

Format de message d'état de l'agent MFT

Lorsqu'un Managed File Transfer Agent est créé ou démarré, l'agent publie ses détails dans SYSTEM.FTE sur son gestionnaire de files d'attente de coordination (dans SYSTEM.FTE/Agents/nom de l'agent).

Les informations suivantes sont incluses :

- Nom de l'agent
- Plateforme sur laquelle l'agent s'exécute
- Description de l'agent (si fournie)
- Gestionnaire de files d'attente de l'agent
- Fuseau horaire dans lequel l'agent s'exécute
- Version d'agent
- Limites de transfert d'agent
- Etat de chacun des transferts en cours de l'agent. Ces états sont répertoriés dans [Etats de transfert d'agent](#)
- Type d'agent

Si l'agent est un agent de pont de protocole, les informations suivantes sont également incluses:

- Type d'agent de pont de protocole
- Nom d'hôte ou adresse IP du serveur de pont de protocole

Le statut de l'agent est republié chaque fois que les états de transfert de l'agent changent, mais par défaut pas plus de 30 secondes. Vous pouvez modifier ce paramètre par défaut à l'aide de la propriété d'agent de limite agentStatusPublishRate, qui est décrite dans: [Propriétés d'agent avancées](#).

L'exemple de sortie suivant montre les clés utilisées pour chaque élément de données dans le statut de l'agent:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE properties SYSTEM "http://java.sun.com/dtd/properties.dtd">
<properties>
  <entry key="agentOsName">Windows 7</entry>
  <entry key="agentDescription"/>
  <entry key="queueManager">QM1</entry>
  <entry key="agentTimeZone">Europe/London</entry>
  <entry key="agentVersion">1.00</entry>
  <entry key="agentName">FTEAGENT</entry>
  <entry key="maxDestinationTransfers">25</entry>
  <entry key="maxSourceTransfers">25</entry>
  <entry key="maxQueuedTransfers">100</entry>
  <entry
key="DestinationTransferStates">414d51204d554e474f202020202020d857374a60a72622=RunningTransfer
414d51204d554e474f202020202020d857374a69a72622=RunningTransfer
414d51204d554e474f202020202020d857374a75a72622=RunningTransfer
  </entry>
  <entry
key="SourceTransferStates">414d51204d554e474f202020202020d857374a93a72622=NegotiatingTransfer
414d51204d554e474f202020202020d857374a78a72622=RunningTransfer
414d51204d554e474f202020202020d857374aaba72622=NewSenderTransfer
414d51204d554e474f202020202020d857374a63a72622=RunningTransfer
  </entry>
</properties>
```

L'exemple de sortie suivant montre les clés utilisées pour chaque élément de données dans le statut de l'agent d'un agent de pont de protocole:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE properties SYSTEM "http://java.sun.com/dtd/properties.dtd">
<properties>
  <entry key="agentOsName">Windows 7</entry>
  <entry key="agentDescription"/>
  <entry key="queueManager">QM1</entry>
  <entry key="agentTimeZone">Europe/London</entry>
  <entry key="agentVersion">1.00</entry>
  <entry key="agentName">BRIDGE</entry>
  <entry key="protocolBridgeType">ftp</entry>
  <entry key="protocolBridgeServerHost">ftpserver.example.org</entry>
  <entry key="maxDestinationTransfers">25</entry>
  <entry key="maxSourceTransfers">25</entry>
  <entry key="maxQueuedTransfers">100</entry>
  <entry key="DestinationTransferStates">414d51204d554e474f202020202020d857374a60a72622=RunningTransfer
  </entry>
  <entry key="SourceTransferStates">414d51204d554e474f202020202020d857374a93a72622=NegotiatingTransfer
  </entry>
</properties>
```

Etats de transfert de l'agent MFT

Un Managed File Transfer Agent démarré publie ses détails dans SYSTEM.FTE sur son gestionnaire de files d'attente de coordination. Ces détails incluent les états de chacun des transferts en cours impliquant cet agent.

Etat de transfert	Explication
Transfert NewSender	Nouveau transfert de l'agent source pour lequel la négociation n'a pas démarré.
Transfert NewReceiver	Un nouveau transfert a été créé sur l'agent de destination dans le cadre de la négociation, mais le transfert n'est pas encore en cours d'exécution.
NegotiatingTransfer	Un agent source est en cours de négociation avec l'agent cible avant d'exécuter un transfert.
RunningTransfer	Un transfert à partir d'un agent source ou d'un agent de destination qui est à l'état d'exécution normal

Etat de transfert	Explication
RecoveringTransfer	Lorsqu'un agent source ou de destination démarre le processus de récupération, tous les transferts à l'état d'exécution passent à l'état de transfert. Les transferts passent de cet état à l'état ReSynchronising lorsqu'un message de resynchronisation est envoyé à l'agent homologue. Par exemple, si l'agent de destination démarre le processus de récupération pour un transfert en cours d'exécution, le transfert passe à l'état de transfert ReSynchronising lorsqu'un message de resynchronisation est envoyé à son agent source.
ReSynchronising	Un agent source ou de destination de transfert a détecté un problème et a envoyé un message de resynchronisation à son agent source ou de destination respectif.
CompletedTransfer	Un agent de destination a terminé le transfert et a envoyé un message d'achèvement à l'agent source. L'agent de destination attend un message d'accusé de réception de l'agent source.
Transfert CompleteReceived	Un agent source a reçu un message d'achèvement de l'agent de destination et a envoyé un message à l'agent de destination pour accuser réception du message d'achèvement.
Transfert CancelledNew	Un agent source a reçu un message d'annulation pour un nouveau transfert.
CancelledInProgressTransfer	Un agent source a reçu un message d'annulation pour un transfert en cours.
ResumingTransfer	Un agent source a reçu un message de réponse de resynchronisation et planifie maintenant le redémarrage du transfert.
RestartingTransfer	Un agent source ou de destination a reçu un message de demande de resynchronisation et attend le redémarrage de l'agent source ou de destination concerné.
WaitingForDestinationCapacity	Un agent source a reçu une erreur DESTINATION_CAPACITY_EXCEEDED de la part de l'agent cible. Le transfert est maintenant en attente d'une nouvelle tentative après un certain temps.
FailedTransferArrêt	Le transfert a échoué mais le message du journal d'achèvement n'a pas été publié et le transfert n'a pas été supprimé du magasin d'état. Par exemple, cet état peut se produire si un processus d'agent est arrêté après la réception d'une réponse d'échec de l'agent de destination mais avant la fin du traitement suivant.

Format des messages de la liste de surveillance MFT

Les messages XML qui sont publiés en tant que publications conservées dans la chaîne de rubrique SYSTEM.FTE/monitors/nom_agent/nom_moniteur sont conformes au schéma MonitorList.xsd. Chaque message XML répertorie un moniteur actif appartenant à cet agent. Ces informations sont utilisées par la commande **fteListMonitors** et le plug-in IBM MQ Explorer pour afficher la liste des moniteurs à l'utilisateur. Le document de schéma MonitorList.xsd se trouve dans le répertoire MQ_INSTALLATION_PATH/mqft/samples/schema. Le MonitorList.xsd schéma importe Monitor.xsd, qui se trouve dans le même répertoire.

Schéma

Le schéma suivant décrit les éléments valides dans un message XML de liste de surveillance.

```
<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema"
targetNamespace="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition"
xmlns="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition">

  <xsd:include schemaLocation="Monitor.xsd"/>

  <xsd:element name="monitorList">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="status" type="monitorStatusType" minOccurs="1" maxOccurs="1"/>
        <xsd:element name="configuration" type="monitorConfigurationType" minOccurs="1" maxOccurs="1"/>
        <xsd:element name="pollInterval" type="pollIntervalType" minOccurs="1" maxOccurs="1"/>
        <xsd:element name="batch" type="batchType" minOccurs="1" maxOccurs="1"/>
        <xsd:any minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="version" type="versionType" use="required"/>
      <xsd:attribute name="agent" type="xsd:string" use="required"/>
      <xsd:attribute name="monitor" type="xsd:string" use="required"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="monitorStatusType">
    <xsd:sequence>
      <xsd:any minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="state" type="xsd:token"/>
    <xsd:anyAttribute/>
  </xsd:complexType>

  <xsd:complexType name="monitorConfigurationType">
    <xsd:sequence>
      <xsd:element name="description" type="xsd:string" minOccurs="1" maxOccurs="1"/>
      <xsd:element name="resources" type="monitorResourcesType" minOccurs="0" maxOccurs="1"/>
      <xsd:element name="triggerMatch" type="triggerMatchType" minOccurs="0" maxOccurs="1"/>
      <xsd:element name="tasks" type="monitorListTasksType" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
    <xsd:anyAttribute/>
  </xsd:complexType>

  <xsd:complexType name="monitorListTasksType">
    <xsd:sequence>
      <xsd:element name="task" type="monitorListTaskType" minOccurs="1" maxOccurs="1"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="monitorListTaskType">
    <xsd:sequence>
      <xsd:element name="name" type="monitorTaskNameType" minOccurs="0" maxOccurs="1"/>
      <xsd:element name="description" type="xsd:string" minOccurs="0" maxOccurs="1"/>
      <xsd:element name="taskXML" type="xsd:string" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

Description du message de la liste des moniteurs

Les éléments et les attributs utilisés dans les messages de la liste de surveillance sont décrits dans la liste suivante:

<monitorList>

L'élément de groupe contenant les éléments décrit un moniteur défini pour l'agent.

Attribut	Description
Agent	Obligatoire. Nom de l'agent sur lequel le moniteur de ressources est défini.
moniteur	Obligatoire. Nom du moniteur. Unique pour cet agent.
version	Obligatoire. Version du format de message de la liste de surveillance.

<statut>

Statut du moniteur.

Attribut	Description
état	Etat du moniteur.

<configuration>

L'élément de groupe contenant les éléments décrit la configuration du moniteur.

<description>

Description du moniteur. (Non utilisé actuellement.)

<Resources>

Ressource (s) surveillée (s).

<répertoire>

Répertoire à surveiller.

Attribut	Description
recursionLevel	Nombre de niveaux de répertoire inférieurs au niveau supérieur à surveiller.
ID	ID de la ressource.

< file_attente>

Une file d'attente à surveiller.

Attribut	Description
ID	ID de la ressource.

<triggerMatch>

Élément contenant l'élément < conditions>.

<conditions>

Élément qui contient la ou les conditions que le moniteur de ressources surveille. Cet élément ne peut contenir que l'un des éléments suivants: <allOf>, <anyOf> ou < condition>.

<allOf>

Élément qui contient la ou les conditions que le moniteur de ressources surveille. Cet élément peut contenir un ou plusieurs éléments < condition>. Pour que le moniteur de ressources soit déclenché, toutes les conditions à l'intérieur de cet élément doivent être remplies.

<anyOf>

Élément qui contient la ou les conditions que le moniteur de ressources surveille. Cet élément peut contenir un ou plusieurs éléments < condition>. Pour que le moniteur de ressources soit déclenché, une seule des conditions à l'intérieur de cet élément doit être remplie.

< condition>

Élément qui contient une condition unique pour laquelle le moniteur de ressources est en cours de surveillance. Cet élément ne peut contenir que l'un des éléments suivants: <fileMatch>, <fileNoMatch>, <fileSize>, <queueNotEmpty>, <completeGroups> ou <fileSizeSame>. Il peut également contenir un élément < name> et un élément < resource>.

Si la ressource surveillée est un répertoire, l'un des trois éléments suivants doit être spécifié dans la condition:

- fileMatch
- fileNocorrespondance
- fileSize

Si la ressource surveillée est une file d'attente, l'un des deux éléments suivants doit être spécifié dans la condition:

- queueNotvide
- completeGroups

<fileMatch>

Élément de groupe pour une condition de correspondance de nom de fichier.

< modèle >

Indique un modèle de correspondance de nom de fichier. Les fichiers de la ressource doivent correspondre au modèle afin de satisfaire à la condition. Le modèle par défaut est * (tout fichier correspond).

<fileNoCorrespondance >

Élément de groupe pour une condition de correspondance de nom de fichier inverse.

< modèle >

Indique un modèle de correspondance de nom de fichier inverse. Si aucun fichier de la ressource contrôlée ne correspond, la condition est satisfaite. Le modèle par défaut est * (l'absence de fichier correspond).

<fileSize >

Élément de groupe pour une comparaison de taille de fichier.

< compare >

Indique une comparaison de taille de fichier. Cette valeur doit être un nombre entier non négatif.

Attribut	Description
opérateur	Opérateur de comparaison à utiliser. Seul > = est pris en charge.
unités	Indique les unités de taille de fichier, qui peuvent être l'une des suivantes: <ul style="list-style-type: none"> • B-octets • Ko-kilooctets • Mo-mégaoctets • Giga-Go La valeur des unités est insensible à la casse, donc mb fonctionne aussi bien que MB'.

< modèle >

Modèle de nom de fichier à mettre en correspondance. La valeur par défaut est * (tout fichier correspond).

<queueNotvide >

Cela ne peut être spécifié que si la ressource est une file d'attente. Indique qu'un message doit figurer dans la file d'attente pour que le moniteur soit déclenché.

<completeGroups >

Cela ne peut être spécifié que si la ressource est une file d'attente. Indique qu'un groupe complet de messages doit être présent dans la file d'attente pour que le moniteur soit déclenché. Une tâche de transfert unique est exécutée pour chaque groupe complet de la file d'attente.

< nom >

Nom de la condition.

<resource >

Identifie la définition de ressource à laquelle la condition doit être comparée.

Attribut	Description
ID	Identificateur unique de la ressource.

< tâches >

Élément de groupe devant contenir des éléments qui spécifient les tâches à appeler lorsque les conditions de déclenchement du moniteur sont satisfaites.

< tâche >

Élément de groupe qui définit une tâche individuelle que le moniteur appelle lorsque les conditions de déclenchement sont satisfaites. Actuellement, une seule tâche peut être spécifiée.

< nom >

Nom de la tâche. Accepte tous les caractères alphanumériques.

<description >

Description de la tâche. Toute valeur de texte est autorisée.

<taskXML >

Message XML qui décrit la tâche que le moniteur doit effectuer. Le contenu de cet élément est au format XML avec caractères d'échappement.

<pollInterval >

Intervalle de temps entre chaque vérification de la ressource par rapport à la condition de déclenchement.

Attribut	Description
unités	Indique les unités de temps de l'intervalle d'interrogation. Les valeurs admises sont : <ul style="list-style-type: none"> • secondes • minutes • heures • jours • semaines • mois • années

< lot >

Nombre maximal de correspondances de déclencheur à inclure dans un lot unique.

Attribut	Description
maxSize	Nombre maximal de correspondances de déclencheur à inclure dans un lot unique

Le code XML suivant montre un exemple de publication conservée qui est publiée dans la chaîne de rubrique SYSTEM.FTE/monitors/nom_agent/MONITORTWO lorsque le moniteur appelé MONITORTWO est créé sur AGENT_JUPITER. Le code XML avec caractères d'échappement dans l'élément <taskXML> décrit la tâche qui est soumise lorsque la condition du moniteur est remplie.

```
<?xml version="1.0" encoding="UTF-8"?>
<lst:monitorList xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xmlns:lst="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition"
xsi:schemaLocation="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition MonitorList.xsd"
version="4.00"
agent="AGENT_JUPITER"
monitor="MONITORTWO">
<status state="started"/>
<configuration>
<description/>
<resources>
<directory recursionLevel="0" id="">/srv/nfs/incoming</directory>
</resources>
<triggerMatch>
<conditions>
<condition>
<name/>
<resource id=""/>
<fileMatch>
<pattern>*.completed</pattern>
</fileMatch>
</condition>
</conditions>
</triggerMatch>
<tasks>
<task>
<name/>
<description/>
<taskXML>&lt;?xml version="1.0" encoding="UTF-8"?&gt;&lt;request
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance" version="4.00"
xsi:noNamespaceSchemaLocation="FileTransfer.xsd"&gt;&lt;managedTransfer&gt;
&lt;originator&gt;&lt;hostName&gt;example.com.&lt;/hostName&gt;
&lt;userID&gt;mqm&lt;/userID&gt;&lt;/originator&gt;
```

```

        &lt;sourceAgent QMgr="QM_JUPITER" agent="AGENT_JUPITER"/&gt;
        &lt;destinationAgent QMgr="QM_JUPITER" agent="AGENT_SATURN"/&gt;
        &lt;transferSet&gt;&lt;item checksumMethod="MD5" mode="binary"&gt;
        &lt;source disposition="leave" recursive="false"&gt;&lt;file
        &lt;/src/nfs/incoming/*.txt&lt;/file&gt;&lt;/source&gt;
        &lt;destination exist="error" type="directory"&gt;
        &lt;/file&gt;/srv/backup&lt;/file&gt;&lt;/destination&gt;
        &lt;/item&gt;&lt;/transferSet&gt;&lt;/managedTransfer&gt;
        &lt;/request&gt;
    &lt;/taskXML>
</task>
</tasks>
</configuration>
<pollInterval units="minutes">1</pollInterval>
<batch maxSize="1"/>
</lst:monitorList>

```

Format de message de la liste de planification MFT

Message XML publié dans une publication conservée dans la chaîne de rubrique SYSTEM.FTE/Scheduler/nom_agent est conforme au schéma ScheduleList.xsd. Ce message XML répertorie toutes les planifications actives appartenant à cet agent. Ces informations sont utilisées par la commande **ftelListScheduledTransfers** et par IBM MQ Explorer pour afficher la liste des planifications à l'utilisateur. Le document de schéma ScheduleList.xsd se trouve dans le répertoire *MQ_INSTALLATION_PATH/mqft/samples/schema*. Le ScheduleList.xsd schéma importe FileTransfer.xsd, qui se trouve dans le même répertoire.

Schéma

Le schéma suivant décrit les éléments valides dans un message XML de liste de surveillance.

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema">
  <xsd:include schemaLocation="FileTransfer.xsd"/>
  <xsd:element name="schedules">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="managedTransfer" type="scheduledManagedTransferType" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute name="version" type="versionType" use="required"/>
      <xsd:attribute name="size" type="xsd:nonNegativeInteger" use="required"/>
      <xsd:attribute name="agent" type="xsd:string" use="required"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:complexType name="scheduledManagedTransferType">
    <xsd:sequence>
      <xsd:element name="originator" type="origRequestType" maxOccurs="1" minOccurs="1"/>
      <xsd:element name="schedule" type="scheduleListType" maxOccurs="1" minOccurs="0"/>
      <xsd:element name="sourceAgent" type="agentType" maxOccurs="1" minOccurs="1"/>
      <xsd:element name="destinationAgent" type="agentClientType" maxOccurs="1" minOccurs="1"/>
      <xsd:element name="trigger" type="triggerType" maxOccurs="1" minOccurs="0"/>
      <xsd:element name="reply" type="replyType" maxOccurs="1" minOccurs="0"/>
      <xsd:element name="transferSet" type="transferSetType" maxOccurs="1" minOccurs="1"/>
      <xsd:element name="job" type="jobType" maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="id" type="idType" use="required"/>
  </xsd:complexType>
  <xsd:complexType name="scheduleListType">
    <xsd:sequence>
      <xsd:element name="submit" type="submitType" maxOccurs="1" minOccurs="1"/>
      <xsd:element name="repeat" type="repeatType" maxOccurs="1" minOccurs="0"/>
      <xsd:element name="next" type="noZoneTimeType" maxOccurs="1" minOccurs="1"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>

```

Description du message de la liste de planification

Les éléments et attributs utilisés dans les messages de liste de planification sont décrits dans la liste suivante:

< planifications >

Élément de groupe contenant des informations sur toutes les planifications définies sur un seul agent.

Attribut	Description
Agent	Obligatoire. Nom de l'agent source sur lequel la planification est définie.
taille	Obligatoire. Nombre de planifications définies sur cet agent.
version	Obligatoire. Version du format de message de la liste de planification.

<managedTransfer>

Élément de groupe contenant des informations sur une seule planification.

Attribut	Description
ID	Obligatoire. ID chaîne hexadécimale du message de demande de planification.

<originaire>

Émetteur de la demande de planification.

<hostName>

Nom d'hôte de la machine à partir de laquelle la demande de planification a été soumise.

<userID>

ID de l'utilisateur qui a soumis la demande de planification.

<IDmqmdUser>

ID utilisateur MQMD de l'utilisateur qui a soumis la demande de planification.

< planification >

Élément contenant les éléments qui décrivent le moment où le transfert planifié a lieu.

< soumettre >

Indique la date et l'heure auxquelles le transfert planifié doit commencer.

Attribut	Description
base de temps	Indique le fuseau horaire à utiliser. La valeur de cet attribut peut être l'une des valeurs suivantes: <ul style="list-style-type: none"> • source-utiliser le fuseau horaire de l'agent source • admin-utilisez le fuseau horaire de l'administrateur qui a émis la commande • UTC-utiliser le temps universel coordonné
fuseau horaire	Description du fuseau horaire en fonction de la valeur de base de temps

< répétition >

Élément de groupe qui contient des détails sur la fréquence de répétition d'un transfert planifié, le nombre de répétitions d'un transfert planifié et le moment où un transfert planifié cesse de se répéter.

Attribut	Description
intervalle	Les unités d'intervalle, qui doivent être l'une des valeurs suivantes: <ul style="list-style-type: none"> • minutes • heures • jours • semaines • mois • années

< fréquence >

Période qui doit s'écouler avant que le transfert ne se répète.

Attribut	Description
intervalle	Les unités d'intervalle, qui doivent être l'une des valeurs suivantes: <ul style="list-style-type: none"> • minutes

Attribut	Description
	<ul style="list-style-type: none"> • heures • jours • semaines • mois • années

<expireTime>

Élément facultatif indiquant la date et l'heure d'arrêt d'un transfert planifié répétitif. Cet élément et l'élément <expireCount> s'excluent mutuellement.

<expireCount>

Élément facultatif qui indique le nombre de fois où le transfert de fichiers planifié se produit avant l'arrêt. Cet élément et l'élément <expireTime> s'excluent mutuellement.

< next>

Indique la date et l'heure auxquelles le prochain transfert planifié doit commencer.

<sourceAgent>

Indique le nom de l'agent sur le système où se trouve le fichier source.

Attribut	Description
Agent	Indique le nom de l'agent.
QMgr	Nom du gestionnaire de files d'attente d'agent.

<destinationAgent>

Indique le nom de l'agent sur le système vers lequel vous souhaitez transférer le fichier.

Attribut	Description
Agent	Indique le nom de l'agent.
QMgr	Nom du gestionnaire de files d'attente d'agent.

< déclencheur>

Élément facultatif qui spécifie une condition qui doit être vraie pour que le transfert de fichier ait lieu.

Attribut	Description
journal	Indicateur signalant si les échecs de déclenchement sont consignés. Les valeurs admises sont les suivantes: <ul style="list-style-type: none"> • yes-des entrées de journal sont créées pour les transferts déclenchés ayant échoué • no-les entrées de journal ne sont pas créées pour les transferts déclenchés ayant échoué

< réponse>

Indique le nom de la file d'attente de réponses temporaire générée pour les transferts de fichiers synchrones (indiquée avec le paramètre **-w** sur la ligne de commande). Le nom de la file d'attente est défini par la clé **dynamicQueuePrefix** dans le fichier de configuration `command.properties` ou par la valeur par défaut `WMQFTE.*` Si aucune valeur n'est spécifiée,

Attribut	Description
QMGR	Nom du gestionnaire de files d'attente de commandes sur lequel la file d'attente dynamique temporaire est générée pour recevoir les réponses.

<transferSet>

Indique un groupe de transferts de fichiers que le transfert planifié doit effectuer ensemble. Lors de la transmission, <transferSet> est un élément de groupe contenant des éléments < item>.

Attribut	Description
priorité	Niveau de priorité du transfert. La priorité est une valeur comprise entre 0 et 9, où 0 est la priorité la plus basse. Le niveau de priorité par défaut est 0 et, par défaut, le transfert utilise le niveau de priorité de l'agent source.

< travail>

Élément de groupe facultatif contenant des informations de travail pour l'ensemble de la spécification de transfert. < job> est un identificateur de nom de travail défini par l'utilisateur qui est ajouté au message du journal lorsque le transfert a commencé. Cet élément < job> est identique à l'élément < job> qui apparaît dans le message du journal de transfert, décrit dans la rubrique suivante: [«Formats des messages du journal de transfert de fichiers»](#), à la page 771.

Exemple

```
<?xml version="1.0" encoding="UTF-8"?>
<schedules xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  size="2"
  version="4.00"
  agent="AGENT_JUPITER"
  xsi:noNamespaceSchemaLocation="ScheduleList.xsd">
  <managedTransfer id="1">
    <originator>
      <hostName>example.com.</hostName>
      <userID>mqm</userID>
    </originator>
    <schedule>
      <submit timebase="admin" timezone="Europe/London">2010-01-01T21:00+0000</
submit>
      <next>2010-01-01T21:00+0000</next>
    </schedule>
    <sourceAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER"/>
    <destinationAgent agent="AGENT_SATURN" QMgr="QM_JUPITER"/>
    <reply QMGR="QM_JUPITER">WMQFTE.4D400F8B20004E06</reply>
    <transferSet>
      <item mode="binary" checksumMethod="MD5">
        <source recursive="false" disposition="leave">
          <file>/etc/passwd</file>
        </source>
        <destination type="directory" exist="overwrite">
          <file>/tmp</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
  <managedTransfer id="2">
    <originator>
      <hostName>example.com.</hostName>
      <userID>mqm</userID>
    </originator>
    <schedule>
      <submit timebase="admin" timezone="Europe/London">2010-12-31T09:00+0000</
submit>
      <next>2010-12-31T09:00+0000</next>
    </schedule>
    <sourceAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER"/>
    <destinationAgent agent="AGENT_NEPTUNE" QMgr="QM_JUPITER"/>
    <reply QMGR="QM_JUPITER">WMQFTE.4D400F8B20004E09</reply>
    <transferSet>
      <item mode="binary" checksumMethod="MD5">
        <source recursive="false" disposition="leave">
          <file>/etc/passwd</file>
        </source>
        <destination type="directory" exist="overwrite">
          <file>/tmp</file>
        </destination>
      </item>
    </transferSet>
```

```
</managedTransfer>
</schedules
```

Exemple de message XML de modèle MFT

Lorsqu'un modèle est créé, un message est publié dans SYSTEM.FTE avec une chaîne de rubrique Templates/*template_ID*. Cet exemple XML décrit un modèle unique défini dans votre réseau Managed File Transfer .

```
<?xml version="1.0" encoding="UTF-8"?>
<transferTemplate version="4.00" id="baf9df73-45c2-4bb0-a085-292232ab66bc">
  <name>BASIC_TEMPLATE</name>
  <sourceAgentName>AGENT_JUPITER</sourceAgentName>
  <sourceAgentQMgr>QM_JUPITER</sourceAgentQMgr>
  <destinationAgentName>AGENT_SATURN</destinationAgentName>
  <destinationAgentQMgr>QM_JUPITER</destinationAgentQMgr>
  <fileSpecs>
    <item mode="binary" checksumMethod="MD5">
      <source recursive="false" disposition="leave">
        <file>/etc/passwd</file>
      </source>
      <destination type="directory" exist="overwrite">
        <file>/tmp</file>
      </destination>
    </item>
  </fileSpecs>
  <priority>0</priority>
</transferTemplate>
```

Tâches associées

«Création d'un modèle de transfert de fichiers à l'aide de IBM MQ Explorer», à la page 202
Vous pouvez créer un modèle de transfert de fichier à partir de IBM MQ Explorer ou de la ligne de commande. Vous pouvez ensuite utiliser ce modèle pour créer de nouveaux transferts de fichiers à l'aide des détails du modèle ou soumettre le modèle pour démarrer le transfert de fichiers.

Référence associée

«fteCreateTemplate: créer un nouveau modèle de transfert de fichiers», à la page 431

La commande **fteCreateTemplate** crée un modèle de transfert de fichier que vous pouvez conserver pour une utilisation ultérieure. Le seul paramètre obligatoire est le paramètre **-tn** (*template_name*). Tous les autres paramètres sont facultatifs, bien que si vous spécifiez une spécification de fichier source, vous devez également fournir un fichier de destination. De même, si vous spécifiez un fichier de destination, vous devez également spécifier une spécification de fichier source.

Format des messages d'état de transfert de fichiers

Les messages sont publiés dans le gestionnaire de files d'attente de coordination pour indiquer le statut de transfert de chaque fichier de l'ensemble de transferts. Chaque fois qu'une demande de transfert de fichier est traitée par l'agent, un message de transaction est publié dans le gestionnaire de files d'attente de coordination (sur son SYSTEM.FTE/Transfers/*nom_agent/ID_transfert*), qui est conforme au schéma XML TransferStatus.xsd . Le fichier TransferStatus.xsd se trouve dans le répertoire MQ_INSTALLATION_PATH/mqft/samples/schema de votre installation WMQMF.

Schéma

Le schéma suivant décrit les éléments valides dans un message XML de statut de transfert.

```
<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema">
  <xsd:include schemaLocation="fteutils.xsd"/>
  <xsd:element name="transaction">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="sourceAgent" type="agentType"
          maxOccurs="1" minOccurs="1"/>
        <xsd:element name="destinationAgent" type="agentType"
          maxOccurs="1" minOccurs="1"/>
        <xsd:element name="transferSet" type="transferSetType"
```

```

        maxOccurs="1"          minOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute name="version" type="versionType" use="required"/>
    <xsd:attribute name="ID" type="IDType" use="required"/>
</xsd:complexType>
</xsd:element>

<xsd:complexType name="transferSetType">
    <xsd:sequence>
        <xsd:element name="stats" type="statsType"
            maxOccurs="1" minOccurs="1"/>
        <xsd:element name="current" type="currentType"
            maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="time" type="xsd:dateTime" use="required"/>
</xsd:complexType>

<xsd:complexType name="currentType">
    <xsd:sequence>
        <xsd:element name="source" type="fileSourceType"
            maxOccurs="1" minOccurs="1"/>
        <xsd:element name="destination" type="fileDestinationType"
            maxOccurs="1" minOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute name="transferred" type="xsd:nonNegativeInteger"
use="required"/>
    <xsd:attribute name="size" type="xsd:nonNegativeInteger" use="required"/>
</xsd:complexType>

    <xsd:complexType name="statsType">
        <xsd:attribute name="bytes" type="xsd:nonNegativeInteger"
use="required"/>
        <xsd:attribute name="seconds" type="xsd:decimal"
use="required"/>
        <xsd:attribute name="currentItem" type="xsd:nonNegativeInteger"
use="required"/>
        <xsd:attribute name="totalItems" type="xsd:nonNegativeInteger" use="required"/>
    </xsd:complexType>
</xsd:schema>

```

Description du message de statut de transfert

Les éléments et attributs utilisés dans les messages de statut de transfert sont décrits dans la liste suivante:

< transaction >

Élément de groupe qui contient tous les éléments pour les transferts de fichiers.

Attribut	Description
version	Indique la version de cet élément fournie par Managed File Transfer.
ID	Identificateur unique du transfert de fichiers.

< sourceAgent >

Indique le nom de l'agent sur le système où se trouve le fichier source.

Attribut	Description
Agent	Nom de l'agent.
QMgr	Nom du gestionnaire de files d'attente d'agent.

< destinationAgent >

Indique le nom de l'agent sur le système vers lequel vous souhaitez transférer le fichier.

Attribut	Description
Agent	Nom de l'agent.
QMgr	Nom du gestionnaire de files d'attente d'agent.

< ensemble_transfert >

Indique un groupe de transferts de fichiers en cours d'exécution. Tous les fichiers du transfert doivent provenir du même agent source et se terminer par le même agent de destination.

Attribut	Description
heure	Indique la date et l'heure (au format date / heure).

< stats >

Obligatoire. Définit des métriques sur le transfert, y compris le nombre d'octets copiés jusqu'à présent, en nombre de secondes indiqué. Indique également le nombre d'éléments en cours sur le nombre total d'éléments dans <transferSet>.

Attribut	Description
octets	Nombre d'octets copiés jusqu'à présent.
secondes	Nombre de secondes nécessaires pour transférer ces octets.
currentItem	Index de l'élément en cours de transfert.
totalItems	Nombre total d'éléments transférés.

< Actuel >

Élément facultatif. Élément de groupe contenant des éléments qui spécifient le transfert de fichier en cours. L'élément < current > indique le nombre d'octets de données transférés jusqu'à présent pour l'élément en cours et le nombre total d'octets attendu

< source >

Élément de groupe qui contient l'élément spécifiant le nom de fichier source.

< fichier >

Indique le chemin source du fichier en cours de transfert. Le chemin d'accès est celui indiqué pour le transfert. Ce chemin peut être différent du chemin qui est généré dans le journal de transfert, qui est la forme absolue du chemin.

< destination >

Élément de groupe qui contient l'élément spécifiant le nom ou la spécification du fichier de destination.

< fichier >

Indique le chemin de destination du fichier en cours de transfert. Le chemin d'accès est celui indiqué pour le transfert. Ce chemin peut être différent du chemin qui est généré dans le journal de transfert, qui est la forme absolue du chemin.

Attribut	Description
alias	Indique un alias pour le fichier de destination. Cet alias est le nom du fichier source, à l'exclusion de tout chemin de répertoire spécifié pour le transfert.
espace fichier	Indique le nom de l'espace fichier dans lequel le fichier de destination est écrit.

< file_attente >

Lorsqu'il est utilisé avec l'élément < destination >, indique le nom de la file d'attente vers laquelle vous souhaitez effectuer le transfert. Ce nom est au format QUEUE ou QUEUE@QUEUE_MANAGER.

Exemples de message de progression de transfert de fichier

Lorsqu'un transfert est en cours, des messages sont publiés dans SYSTEM.FTE avec une chaîne de rubrique Transfers/nom_agent/ID_transfert. Les exemples XML montrent le message de progression pour un transfert de fichier unique et pour un transfert de fichier multiple.

Transfert de fichier unique

L'exemple suivant montre les détails d'un transfert de fichier unique en cours.

```
<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  ID="414d51205553322e42494e44494e47538b0f404d223d0020"
  xsi:noNamespaceSchemaLocation="TransferStatus.xsd">
  <sourceAgent agent="US2.BINDINGS.FILE" QMgr="US2.BINDINGS"/>
  <destinationAgent agent="US2.BINDINGS.FILE" QMgr="US2.BINDINGS"/>
  <transferSet time="2011-01-26T13:03:26.542Z">
  <stats bytes="1198" seconds="0.018" currentItem="1" totalItems="1"/>
  <current transferred="1151" size="1151">
    <source>
      <file>/etc/passwd</file>
    </source>
    <destination>
      <file>/tmp/passwd</file>
    </destination>
  </current>
</transferSet>
</transaction>
```

Transfert de fichiers multiples

S'il y a plus de fichiers dans l'ensemble de transfert, le message d'état de transfert indique lequel est en cours de traitement et combien d'octets ont été transférés jusqu'à présent.

```
<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  ID="414d51205553322e42494e44494e47538b0f404d035c0020"
  xsi:noNamespaceSchemaLocation="TransferStatus.xsd">
  <sourceAgent agent="US2.BINDINGS.FILE" QMgr="US2.BINDINGS"/>
  <destinationAgent agent="US2.BINDINGS.FILE" QMgr="US2.BINDINGS"/>
  <transferSet time="2011-01-26T13:12:58.636Z">
    <stats bytes="440" seconds="0.082" currentItem="10" totalItems="10"/>
    <current transferred="0" size="0">
      <source>
        <file>/srv/nfs/incoming/file10.txt</file>
      </source>
      <destination>
        <file>/srv/nfs/outgoing/file10.txt</file>
      </destination>
    </current>
  </transferSet>
</transaction>
```

Formats des messages du journal de transfert de fichiers

Les messages de journal de transfert de fichiers sont publiés dans SYSTEM.FTE avec une chaîne de rubrique `Log/nom_agent/ID_transfert`. Ces messages sont conformes au schéma `TransferLog.xsd`, qui se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/schema` de votre installation Managed File Transfer.

Si vous souhaitez surveiller les transferts de fichiers ou collecter des données les concernant, configurez un abonnement à une rubrique générique adaptée aux transferts qui vous intéressent. Exemple :

```
Log/#
```

ou,

```
Log/FTEAGENT/#
```

Cet abonnement peut être durable ou non durable. Les abonnements durables continuent d'exister lorsque la connexion d'une application d'abonnement au gestionnaire de files d'attente est fermée. Les

abonnements non durables existent uniquement tant que la connexion d'une application d'abonnement au gestionnaire de files d'attente reste ouverte.

Schéma

Le schéma suivant décrit les éléments valides dans un message XML de journal de transfert.

```
<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema">
  <xsd:include schemaLocation="fteutils.xsd"/>
  <xsd:element name="transaction">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="action" type="actionType" minOccurs="0"/>
        <xsd:element name="sourceAgent" type="agentExitStatusType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="sourceWebGateway" type="webGatewayType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="sourceWebUser" type="webUserType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="destinationAgent" type="agentExitStatusType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="destinationWebGateway" type="webGatewayType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="destinationWebUser" type="webUserType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="agent" type="agentExitStatusType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="originator" type="origRequestType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="status" type="statusType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="trigger" type="triggerType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="transferSet" type="transferSetType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="job" type="jobType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="scheduleLog" type="scheduleLogType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="statistics" type="statisticsType" maxOccurs="1" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="version" type="versionType" use="required"/>
      <xsd:attribute name="ID" type="IDType" use="required"/>
      <xsd:attribute name="relatedID" type="IDType" use="optional"/>
      <xsd:attribute name="agentRole" type="agentRoleType" use="optional"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:complexType name="agentExitStatusType">
    <xsd:complexContent>
      <xsd:extension base="agentType">
        <xsd:sequence>
          <xsd:element name="startExits" type="exitGroupType" minOccurs="0" maxOccurs="1"/>
          <xsd:element name="endExits" type="exitGroupType" minOccurs="0" maxOccurs="1"/>
          <xsd:element name="systemInfo" type="systemInfoType" minOccurs="0" maxOccurs="1"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="transferSetType">
    <xsd:sequence>
      <xsd:element name="metaDataSet" type="metaDataSetType" maxOccurs="1" minOccurs="0"/>
      <xsd:element name="call" type="callGroupType" maxOccurs="1" minOccurs="0"/>
      <xsd:element name="preSourceCall" type="callGroupType" maxOccurs="1" minOccurs="0"/>
      <xsd:element name="postSourceCall" type="callGroupType" maxOccurs="1" minOccurs="0"/>
      <xsd:element name="preDestinationCall" type="callGroupType" maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

```

        <xsd:element name="postDestinationCall" type="callGroupType"
            maxOccurs="1" minOccurs="0"/>
        <xsd:element name="item" type="itemType"
            maxOccurs="unbounded" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="index" type="xsd:nonNegativeInteger" use="optional"/>
    <xsd:attribute name="size" type="xsd:nonNegativeInteger" use="optional"/>
    <xsd:attribute name="startTime" type="xsd:dateTime" use="required"/>
    <xsd:attribute name="total" type="xsd:nonNegativeInteger" use="required"/>
    <xsd:attribute name="bytesSent" type="xsd:nonNegativeInteger" use="required"/>
</xsd:complexType>

<xsd:complexType name="itemType">
    <xsd:sequence>
        <xsd:element name="source" type="fileSourceChecksumType"
            maxOccurs="1" minOccurs="1"/>
        <xsd:element name="destination" type="fileDestinationChecksumType"
            maxOccurs="1" minOccurs="1"/>
        <xsd:element name="status" type="statusType"
            maxOccurs="1" minOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute name="mode" type="modeType" use="required"/>
</xsd:complexType>

<xsd:complexType name="fileSourceChecksumType">
    <xsd:complexContent>
        <xsd:extension base="fileSourceType">
            <xsd:sequence>
                <xsd:element name="checksum" type="checksumType" minOccurs="0"
maxOccurs="1"/>
            </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="fileDestinationChecksumType">
    <xsd:complexContent>
        <xsd:extension base="fileDestinationType">
            <xsd:sequence>
                <xsd:element name="checksum" type="checksumType"
minOccurs="0" maxOccurs="1"/>
            </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="actionType">
    <xsd:simpleContent>
        <xsd:extension base="actionEnumType">
            <xsd:attribute name="time" type="xsd:dateTime" use="required"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:simpleType name="actionEnumType">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="cancelled"/>
        <xsd:enumeration value="started"/>
        <xsd:enumeration value="progress"/>
        <xsd:enumeration value="completed"/>
        <xsd:enumeration value="malformed"/>
        <xsd:enumeration value="notAuthorized"/>
        <xsd:enumeration value="deleted"/>
    </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="systemInfoType">
    <xsd:attribute name="architecture" type="xsd:string" use="required"/>
    <xsd:attribute name="name" type="xsd:string" use="required"/>
    <xsd:attribute name="version" type="xsd:string" use="required"/>
</xsd:complexType>

<xsd:element name="malformed">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="action" type="actionType"
maxOccurs="1" minOccurs="1"/>
            <xsd:element name="agent" type="agentExitStatusType"
maxOccurs="1" minOccurs="0"/>
            <xsd:element name="status" type="statusType"
maxOccurs="1" minOccurs="1"/>
        </xsd:sequence>
    </xsd:complexType>

```

```

        <xsd:attribute name="version" type="versionType" use="required"/>
        <xsd:attribute name="ID" type="IDType" use="required"/>
        <xsd:attribute name="agentRole" type="agentRoleType" use="required"/>
    </xsd:complexType>
</xsd:element>

<xsd:element name="notAuthorized">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="action" type="actionType"
                maxOccurs="1" minOccurs="1"/>
            <xsd:element name="originator" type="origRequestType"
                maxOccurs="1" minOccurs="1"/>
            <xsd:element name="authority" type="xsd:string"
                minOccurs="1" maxOccurs="1"/>
            <xsd:element name="status" type="statusType"
                maxOccurs="1" minOccurs="1"/>
        </xsd:sequence>
        <xsd:attribute name="version" type="versionType" use="required"/>
        <xsd:attribute name="ID" type="IDType" use="required"/>
        <xsd:attribute name="agentRole" type="agentRoleType" use="required"/>
    </xsd:complexType>
</xsd:element>

<xsd:complexType name="statisticsType">
    <xsd:sequence>
        <xsd:element name="actualStartTime" type="xsd:dateTime"
            maxOccurs="1" minOccurs="0"/>
        <xsd:element name="retryCount" type="xsd:nonNegativeInteger"
            maxOccurs="1" minOccurs="1"/>
        <xsd:element name="numFileFailures" type="xsd:nonNegativeInteger"
            maxOccurs="1" minOccurs="1"/>
        <xsd:element name="numFileWarnings" type="xsd:nonNegativeInteger"
            maxOccurs="1" minOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="webGatewayType">
    <xsd:attribute name="webGatewayName" type="xsd:string" use="optional"/>
    <xsd:attribute name="webGatewayAgentName" type="xsd:string" use="optional"/>
    <xsd:attribute name="webGatewayAgentQMgr" type="xsd:string" use="optional"/>
</xsd:complexType>

<xsd:complexType name="webUserType">
    <xsd:attribute name="webGatewayName" type="xsd:string" use="required"/>
    <xsd:attribute name="webGatewayAgentName" type="xsd:string" use="optional"/>
    <xsd:attribute name="webGatewayAgentQMgr" type="xsd:string" use="optional"/>
</xsd:complexType>
</xsd:schema>

```

Remarque : Depuis la IBM MQ 9.0, Managed File Transfer ne prend pas en charge la passerelle Web ou les agents Web.

Description du message du journal de transfert

< transaction >

Élément de groupe qui spécifie un groupe de transferts que vous souhaitez effectuer ensemble.

Attribut	Description
version	Indique la version de cet élément, telle que détaillée par Managed File Transfer.
ID	Indique l'ID transaction unique. L'ID peut comporter jusqu'à 48 caractères alphanumériques.
relatedID	Facultatif. Si la transaction est la suppression ou le téléchargement d'un fichier à partir d'un espace fichier, relatedID indique l'ID de transaction du transfert qui a transféré le fichier dans l'espace fichier.
agentRole	Facultatif. Indique si l'agent concerné se trouve sur le système source ou cible

Attribut	Description
xmlns:xsi	Déclaration d'espace de nom. Indique que les éléments et les types de données utilisés dans ce schéma dérivent de l'espace de nom "https://www.w3.org/2001/XMLSchema-instance".
xsi:noNamespaceSchemaLocation	Indique le nom et l'emplacement du document de schéma XML par rapport auquel valider ce message s'il n'y a pas de déclaration d'espace de nom. La valeur que vous spécifiez pour cet attribut doit faire référence à un document Managed File Transfer TransferLog.xsd.

<action>

Décrit le statut du transfert de fichier à l'heure consignée par l'attribut d'heure. Le statut peut prendre l'une des valeurs suivantes :

- démarré
- en cours
- terminé
- annulé
- syntaxiquement incorrect (indique que le contenu du message de demande de transfert de fichier ne peut pas être interprété).
- notAuthorized
- supprimé

Attribut	Description
heure	Heure à laquelle le statut de transfert a été capturé, exprimée au format UTC.

<sourceAgent>

Indique le nom de l'agent sur le système où se trouve le fichier source. Seul <sourceAgent> ou <sourceWebUser > peut être spécifié.

<startExits>

Élément de groupe contenant un ou plusieurs éléments d'exit utilisateur. Cet élément ne peut apparaître qu'une seule fois.

<endExits>

Élément de groupe contenant un ou plusieurs éléments d'exit utilisateur. Cet élément ne peut apparaître qu'une seule fois.

<systemInfo>

Décrit l'architecture, le nom et la version du système. Cet élément ne peut apparaître qu'une seule fois.

Attribut	Description
Agent	Nom de l'agent sur le système source.
QMgr	Nom du gestionnaire de files d'attente sur le système source.
agentType	Type de l'agent. Les valeurs admises sont : <ul style="list-style-type: none"> • STANDARD-un agent normal • BRIDGE-Agent de pont de protocole • CD_BRIDGE-Un agent de pont Connect:Direct • EMBEDDED-un agent imbriqué • SFG-agent intégré Sterling File Gateway

Attribut	Description
bridgeURL	Facultatif. Si l'agent est un agent de pont de protocole, nom d'hôte du système hébergeant le serveur de protocole.
Pnode	Facultatif. Si l'agent est un agent de pont Connect:Direct , nom du noeud principal Connect:Direct impliqué dans le transfert.
SNODE	Facultatif. Si l'agent est un agent de pont Connect:Direct , nom du noeud secondaire Connect:Direct impliqué dans le transfert.
bridgeNode	Facultatif. Si l'agent est un agent de pont Connect:Direct , nom du noeud Connect:Direct faisant partie du pont Connect:Direct . Il s'agit du même noeud que le noeud principal ou le noeud secondaire.

<destinationAgent>

Indique le nom de l'agent sur le système vers lequel le fichier a été transféré. Vous pouvez indiquer <destinationAgent> ou <destinationWebUser >.

Attribut	Description
Agent	Nom de l'agent sur le système de destination.
QMgr	Nom du gestionnaire de files d'attente sur le système de destination.
agentType	Type de l'agent. Les valeurs admises sont : <ul style="list-style-type: none"> • STANDARD-un agent normal • BRIDGE-Agent de pont de protocole • CD_BRIDGE-Un agent de pont Connect:Direct • EMBEDDED-un agent imbriqué • SFG-agent intégré Sterling File Gateway
bridgeURL	Facultatif. Si l'agent est un agent de pont de protocole, nom d'hôte du système hébergeant le serveur de protocole.
Pnode	Facultatif. Si l'agent est un agent de pont Connect:Direct , nom du noeud principal Connect:Direct impliqué dans le transfert.
SNODE	Facultatif. Si l'agent est un agent de pont Connect:Direct , nom du noeud secondaire Connect:Direct impliqué dans le transfert.
bridgeNode	Facultatif. Si l'agent est un agent de pont Connect:Direct , nom du noeud Connect:Direct faisant partie du pont Connect:Direct . Il s'agit du même noeud que le noeud principal ou le noeud secondaire.

<startExits>

Élément de groupe contenant un ou plusieurs éléments d'exit utilisateur. Cet élément ne peut apparaître qu'une seule fois.

<endExits>

Élément de groupe contenant un ou plusieurs éléments d'exit utilisateur. Cet élément ne peut apparaître qu'une seule fois.

<systemInfo>

Décrit l'architecture, le nom et la version du système. Cet élément ne peut apparaître qu'une seule fois.

<originaire>

Élément de groupe qui contient les éléments spécifiant l'émetteur de la demande.

<hostName>

Nom d'hôte du système sur lequel se trouve le fichier source.

<userID>

ID utilisateur à l'origine du transfert de fichiers.

< IDmqmdUser>

ID utilisateur IBM MQ fourni dans le descripteur de message (MQMD)

< IDwebUser>

Facultatif. ID utilisateur fourni au navigateur Web qui soumet la demande de transfert.

<webBrowser>

Facultatif. Navigateur Web à partir duquel la demande de transfert a été soumise.

< statut>

Code de résultat et messages complémentaires.

< déclencheur>

Élément de groupe contenant les éléments de déclencheur définis dans la demande de transfert d'origine. Ces éléments peuvent être l'un des éléments suivants ou les deux:

<fileExist>

Condition de déclenchement basée sur l'existence ou non d'un fichier

<fileSize>

Condition de déclenchement selon qu'un fichier correspond ou dépasse la taille spécifiée

<transferSet>

Indique un groupe de transferts de fichiers que vous souhaitez effectuer ensemble. Lors de la transmission, <transferSet> est un élément de groupe contenant des éléments < item>.

Attribut	Description
startTime	Enregistre l'heure à laquelle l'ensemble de transferts a commencé, exprimée au format UTC.
total	Indique le nombre total d'éléments de cet ensemble de transferts.
index	Attribut facultatif. Indique la position du premier élément en cours de l'ensemble de transfert. L'attribut d'index est incrémenté de zéro. Par exemple, si l'index est défini sur 1, le message de progression est le deuxième des deux éléments.
taille	Attribut facultatif. Indique le nombre d'éléments dans le rapport de progression.
priorité	Attribut facultatif. Niveau de priorité du transfert. La priorité est une valeur comprise entre 0 et 9, où 0 est la priorité la plus basse. Le niveau de priorité par défaut est 0 et par défaut, le transfert utilise le niveau de priorité de l'agent source.

< Ensemble de métadonnéesmetaData>

Élément de groupe contenant un ou plusieurs des attributs suivants:

<metaData>

Attribut	Description
clé	Moitié de clé d'une paire clé-valeur de métadonnées. Le contenu de l'élément <metaData> contient la moitié de la valeur de la paire. Par exemple, <metaData key="testkey1">testvalue1</metaData>

< travail>

Élément de groupe qui contient un élément spécifiant les détails du travail. < job> est un identificateur de nom de travail défini par l'utilisateur qui est ajouté au message du journal lorsque le transfert a commencé. Cet élément < job> est identique à l'élément < job> inclus dans le message de demande de transfert, décrit dans la rubrique suivante: «Format de message de demande de transfert de fichier», à la page 804.

< nom >

La valeur du nom peut être n'importe quelle chaîne.

<scheduleLog>

Élément de groupe qui contient des éléments spécifiant les noms et les emplacements des fichiers source et de destination.

Attribut	Description
ID	Correspond à l'ID de planification si le transfert est planifié.

<item>

Élément de groupe qui contient des éléments spécifiant les noms et les emplacements des fichiers source et de destination.

< source >

Élément de groupe qui contient l'élément < file> ou l'élément < queue> et l'élément < checksum> du fichier sur le système source.

Attribut	Description
recursive	Indique que les fichiers sont transférés de manière récursive dans des sous-répertoires lorsque l'élément < source> est un répertoire ou contient des caractères génériques.
disposition	Indique l'action effectuée sur l'élément < source> lorsque < source> a été transféré vers sa destination. Les options valides sont les suivantes: <ul style="list-style-type: none"> • leave-les fichiers source restent inchangés. • delete-les fichiers source sont supprimés du système source après le transfert du fichier source.
correlationBoolean	Valeur de corrélation booléenne. Si la source est un pont Connect:Direct , indique si le processus Connect:Direct est défini par l'utilisateur.
correlationString1	Valeur de corrélation de chaîne. Si la source est un pont Connect:Direct , indique le nom du processus Connect:Direct qui se produit à la destination du transfert.
correlationNum1	Valeur de corrélation numérique. Si la source est un pont Connect:Direct , indique le numéro d'ID du processus Connect:Direct qui se produit à la destination du transfert.

< file_attente >

Lorsqu'il est utilisé avec l'élément < source>, indique le nom de la file d'attente à partir de laquelle les messages transférés ont été lus, qui se trouve sur le gestionnaire de files d'attente de l'agent source.

Attribut	Description
messageCount	Nombre de messages lus dans la file d'attente.
groupId	ID groupe IBM MQ des messages lus dans la file d'attente.

< destination >

Élément de groupe qui contient l'élément < file> ou l'élément < queue> et l'élément < checksum> pour la destination.

Un seul des éléments < file> et < queue> est présent en tant qu'élément enfant de la destination.

Attribut	Description
type	Type de destination. Les options valides sont les suivantes: <ul style="list-style-type: none"> • queue-spécifie une file d'attente IBM MQ comme destination • file-spécifie un fichier comme destination • directory-spécifie un répertoire comme destination •  dataset-spécifie un fichier z/OS comme destination •  pds-spécifie un fichier partitionné z/OS comme destination <p>La file d'attente d'options peut être présente uniquement lorsque l'élément < destination > a un élément enfant < queue >. Les autres options ne peuvent être présentes que lorsque l'élément < destination > a un élément enfant de < fichier >.</p>
existe	Indique l'action effectuée si un fichier de destination existe sur le système de destination. Les options valides sont les suivantes: <ul style="list-style-type: none"> • error-signale une erreur et le fichier n'est pas transféré. • overwrite-écrase le fichier de destination existant. <p>Cet attribut ne peut pas être présent si l'élément < destination > a un élément enfant de < queue >.</p>
correlationBoolean	Valeur de corrélation booléenne. Si la destination est un pont Connect:Direct , indique si le processus Connect:Direct est défini par l'utilisateur.
correlationString1	Valeur de corrélation de chaîne. Si la destination est un pont Connect:Direct , indique le nom du processus Connect:Direct qui se produit à la destination du transfert.
correlationNum1	Valeur de corrélation numérique. Si la destination est un pont Connect:Direct , indique le numéro d'ID du processus Connect:Direct qui se produit à la destination du transfert.

<fichier>

Indique le chemin d'accès absolu du fichier qui a été transféré (à la fois à la source et à la destination). Le chemin d'accès complet est dans un format compatible avec votre système d'exploitation, par exemple C : /from/here . txt. Les URI de fichier ne sont pas utilisés.

< file_attente >

Lorsqu'il est utilisé avec l'élément < destination >, indique le nom de la file d'attente vers laquelle le transfert a été effectué, qui se trouve sur tout gestionnaire de files d'attente connecté au gestionnaire de files d'attente de l'agent de destination.

Attribut	Description
messageCount	Nombre de messages écrits dans la file d'attente.
messageLength	Longueur des messages écrits dans la file d'attente.
groupId	Si la demande de transfert indique que le fichier est divisé en plusieurs messages, la valeur de cet attribut correspond à l'ID groupe IBM MQ des messages écrits dans la file d'attente.
messageId	Si la demande de transfert n'indique pas que le fichier est divisé en plusieurs messages, la valeur de cet attribut est l'ID message IBM MQ du message écrit dans la file d'attente.

< checksum >

Élément facultatif.

Indique le type d'algorithme de hachage qui a généré le résumé du message pour créer la signature numérique. Actuellement, Managed File Transfer prend en charge l'algorithme de prétraitement de message 5 (MD5) uniquement. Le total de contrôle vous permet de confirmer que l'intégrité des fichiers transférés est intacte.

< malformé >

Élément de groupe pour les messages syntaxiquement inappropriés.

Attribut	Description
version	
ID	
agentRole	Agent source ou agent de destination

< statistiques >

Élément de groupe pour les informations statistiques pour le transfert (le cas échéant).

< heureactualStart >

Heure réelle à laquelle l'agent a commencé à exécuter le transfert. Typiquement, l'heure est la même que (ou très proche) l'heure de début enregistrée pour le transfert. Toutefois, lorsqu'un agent est occupé, les transferts soumis peuvent être mis en file d'attente jusqu'à ce que l'agent ait la capacité d'exécuter les transferts.

< retryCount >

Nombre de fois où le transfert est passé à l'état de récupération et a été relancé par l'agent. Un transfert peut passer à l'état de reprise car les agents source et de destination perdent la communication, soit en raison d'une erreur réseau IBM MQ, soit parce qu'ils ne reçoivent pas de données ou de messages d'accusé de réception pendant un certain temps. Cette période est déterminée par les propriétés de l'agent: transferAckTimeout and transferAckTimeoutRetries.

< numFileEchecs >

Nombre de fichiers dans transferSet dont le transfert a échoué.

< numFileAvertissements >

Nombre de fichiers dans transferSet qui ont généré des avertissements lors du transfert, mais qui ont été transférés avec succès.

Exemples

Des exemples de messages XML conformes à ce schéma sont fournis pour chacun des types de transfert suivants:

- [Transfert d'un fichier unique](#)
- [Transfert contenant plusieurs fichiers](#)
- [Echec du transfert de fichier](#)
- [Transfert défini avec un déclencheur](#)
- [Transfert démarré par une planification](#)
- [Un transfert qui appelle des exits utilisateur](#)
- [Transfert via un noeud de pont Connect:Direct](#)

Exemples de message de journal de transfert unique

Lorsqu'un transfert est effectué, les messages sont publiés dans SYSTEM.FTE avec une chaîne de rubrique `Log/nom_agent/ID_transfert`. Les exemples XML montrent les messages de journal pour un transfert de fichier unique en cours de démarrage, en cours et terminé.

Transfert de fichier unique-démarré

```
<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  ID="414d51205553322e42494e44494e47538b0f404d223d0020"
  agentRole="sourceAgent"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2011-01-26T13:03:26.484Z">started</action>
  <sourceAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
    <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
  </sourceAgent>
  <destinationAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER"/>
  <originator>
    <hostName>dhcp-9-20-240-199.hursley.ibm.com.</hostName>
    <userID>mqm</userID>
    <mqmdUserID>mqm</mqmdUserID>
  </originator>
  <transferSet startTime="2011-01-26T13:03:26.484Z" total="1" bytesSent="0">
    <metaDataSet>
      <metaData key="com.ibm.wmqfte.SourceAgent">AGENT_JUPITER</metaData>
      <metaData key="com.ibm.wmqfte.DestinationAgent">AGENT_JUPITER</metaData>
      <metaData key="com.ibm.wmqfte.MqmdUser">mqm</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingUser">mqm</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingHost">dhcp-9-20-240-199.hursley.ibm.com.</
metaData>
    <metaData key="com.ibm.wmqfte.TransferId">414d51205553322e42494e44494e47538b0f404d223d0020</
metaData>
    <metaData key="com.ibm.wmqfte.ScheduleId">3</metaData>
    <metaData key="com.ibm.wmqfte.Priority">0</metaData>
  </metaDataSet>
  </transferSet>
  <scheduleLog ID="3"/>
</transaction>
```

Réussite du transfert de fichier unique-progression

```
<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  ID="414d51205553322e42494e44494e47538b0f404d223d0020"
  agentRole="sourceAgent"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2011-01-26T13:03:26.615Z">progress</action>
  <sourceAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
    <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
  </sourceAgent>
  <destinationAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
    <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
  </destinationAgent>
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
    <mqmdUserID>mqm</mqmdUserID>
  </originator>
  <transferSet index="0" size="1" startTime="2011-01-26T13:03:26.484Z" total="1"
bytesSent="1198">
    <item mode="binary">
      <source disposition="leave" type="file">
        <file size="1151" last-modified="2009-11-02T10:37:01.000Z">/etc/passwd</file>
        <checksum method="MD5">2287181c07199f879de28296371cb24c</checksum>
      </source>
      <destination type="file">
        <file size="1151" last-modified="2011-01-26T13:03:26.000Z">/tmp/passwd</file>
        <checksum method="MD5">2287181c07199f879de28296371cb24c</checksum>
      </destination>
      <status resultCode="0"/>
    </item>
  </transferSet>
</transaction>
```

Réussite du transfert de fichier unique-terminé

```
<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  ID="414d51205553322e42494e44494e47538b0f404d223d0020"
  agentRole="sourceAgent"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2011-01-26T13:03:26.622Z">completed</action>
  <sourceAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
    <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
  </sourceAgent>
  <destinationAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
    <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
  </destinationAgent>
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
    <mqmdUserID>mqm</mqmdUserID>
  </originator>
  <status resultCode="0">
    <supplement>BFGRP0032I: The file transfer request has successfully completed.</supplement>
  </status>
  <transferSet startTime="2011-01-26T13:03:26.484Z" total="1" bytesSent="1198">
    <metaDataSet>
      <metaData key="com.ibm.wmqfte.SourceAgent">AGENT_JUPITER</metaData>
      <metaData key="com.ibm.wmqfte.DestinationAgent">AGENT_JUPITER</metaData>
      <metaData key="com.ibm.wmqfte.MqmdUser">mqm</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingUser">mqm</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingHost">example.com.</metaData>
      <metaData key="com.ibm.wmqfte.TransferId">414d51205553322e42494e44494e47538b0f404d223d0020</
metaData>
      <metaData key="com.ibm.wmqfte.ScheduleId">3</metaData>
      <metaData key="com.ibm.wmqfte.Priority">0</metaData>
    </metaDataSet>
  </transferSet>
  <statistics>
    <actualStartTime>2011-01-26T13:03:26.541Z</actualStartTime>
    <retryCount>0</retryCount>
    <numFileFailures>0</numFileFailures>
    <numFileWarnings>0</numFileWarnings>
  </statistics>
</transaction>
```

Exemples de message de journal de transfert de fichiers multiples

Exemples de messages publiés dans SYSTEM.FTE avec une chaîne de rubrique Log/nom_agent/
ID_transfert lorsqu'un transfert contenant plusieurs fichiers se produit.

Transfert de fichiers multiples-démarré

```
<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  ID="414d51205553322e42494e44494e47538b0f404d035c0020"
  agentRole="sourceAgent"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2011-01-26T13:12:58.534Z">started</action>
  <sourceAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
    <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
  </sourceAgent>
  <destinationAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER"/>
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
    <mqmdUserID>mqm</mqmdUserID>
  </originator>
  <transferSet startTime="2011-01-26T13:12:58.534Z" total="6" bytesSent="0">
    <metaDataSet>
      <metaData key="com.ibm.wmqfte.SourceAgent">AGENT_JUPITER</metaData>
      <metaData key="com.ibm.wmqfte.DestinationAgent">AGENT_JUPITER</metaData>
      <metaData key="com.ibm.wmqfte.MqmdUser">mqm</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingUser">mqm</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingHost">example.com.</metaData>
      <metaData key="com.ibm.wmqfte.TransferId">414d51205553322e42494e44494e47538b0f404d035c0020</
metaData>
```

```

    <metaData key="com.ibm.wmqfte.Priority">0</metaData>
  </metaDataSet>
</transferSet>
</transaction>

```

Transfert de fichiers multiples-progression

```

<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  ID="414d51205553322e42494e44494e47538b0f404d035c0020"
  agentRole="sourceAgent"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2011-01-26T13:12:58.753Z">progress</action>
  <sourceAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
    <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
  </sourceAgent>
  <destinationAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
    <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
  </destinationAgent>
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
    <mqmdUserID>mqm</mqmdUserID>
  </originator>
  <transferSet index="0" size="6" startTime="2011-01-26T13:12:58.534Z" total="6" bytesSent="440">
    <item mode="binary">
      <source disposition="leave" type="file">
        <file size="0" last-modified="2011-01-26T13:10:19.000Z"/>/srv/nfs/incoming/file01.txt</
file>
        <checksum method="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>
      </source>
      <destination type="file">
        <file size="0" last-modified="2011-01-26T13:12:58.000Z"/>/srv/nfs/outgoing/file01.txt</
file>
        <checksum method="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>
      </destination>
      <status resultCode="0"/>
    </item>
    <item mode="binary">
      <source disposition="leave" type="file">
        <file size="0" last-modified="2011-01-26T13:10:19.000Z"/>/srv/nfs/incoming/file02.txt</
file>
        <checksum method="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>
      </source>
      <destination type="file">
        <file size="0" last-modified="2011-01-26T13:12:58.000Z"/>/srv/nfs/outgoing/file02.txt</
file>
        <checksum method="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>
      </destination>
      <status resultCode="0"/>
    </item>
    <item mode="binary">
      <source disposition="leave" type="file">
        <file size="0" last-modified="2011-01-26T13:10:19.000Z"/>/srv/nfs/incoming/file03.txt</
file>
        <checksum method="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>
      </source>
      <destination type="file">
        <file size="0" last-modified="2011-01-26T13:12:58.000Z"/>/srv/nfs/outgoing/file03.txt</
file>
        <checksum method="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>
      </destination>
      <status resultCode="0"/>
    </item>
    <item mode="binary">
      <source disposition="leave" type="file">
        <file size="0" last-modified="2011-01-26T13:10:19.000Z"/>/srv/nfs/incoming/file04.txt</
file>
        <checksum method="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>
      </source>
      <destination type="file">
        <file size="0" last-modified="2011-01-26T13:12:58.000Z"/>/srv/nfs/outgoing/file04.txt</
file>
        <checksum method="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>
      </destination>
      <status resultCode="0"/>
    </item>
  </transferSet>
</transaction>

```

```

    <item mode="binary">
      <source disposition="leave" type="file">
        <file size="0" last-modified="2011-01-26T13:10:19.000Z">/srv/nfs/incoming/file05.txt</
file>
        <checksum method="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>
      </source>
      <destination type="file">
        <file size="0" last-modified="2011-01-26T13:12:58.000Z">/srv/nfs/outgoing/file05.txt</
file>
        <checksum method="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>
      </destination>
      <status resultCode="0"/>
    </item>
    <item mode="binary">
      <source disposition="leave" type="file">
        <file size="0" last-modified="2011-01-26T13:10:19.000Z">/srv/nfs/incoming/file06.txt</
file>
        <checksum method="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>
      </source>
      <destination type="file">
        <file size="0" last-modified="2011-01-26T13:12:58.000Z">/srv/nfs/outgoing/file06.txt</
file>
        <checksum method="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>
      </destination>
      <status resultCode="0"/>
    </item>
  </transferSet>
</transaction>

```

Transfert de fichiers multiples-terminé

```

<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  ID="414d51205553322e42494e44494e47538b0f404d035c0020"
  agentRole="sourceAgent"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2011-01-26T13:12:58.766Z">completed</action>
  <sourceAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
    <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
  </sourceAgent>
  <destinationAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
    <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
  </destinationAgent>
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
    <mqmdUserID>mqm</mqmdUserID>
  </originator>
  <status resultCode="0">
    <supplement>BFGRP0032I: The file transfer request has successfully completed.</supplement>
  </status>
  <transferSet startTime="2011-01-26T13:12:58.534Z" total="6" bytesSent="440">
    <metaDataSet>
      <metaData key="com.ibm.wmqfte.SourceAgent">AGENT_JUPITER</metaData>
      <metaData key="com.ibm.wmqfte.DestinationAgent">AGENT_JUPITER</metaData>
      <metaData key="com.ibm.wmqfte.MqmdUser">mqm</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingUser">mqm</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingHost">example.com.</metaData>
      <metaData key="com.ibm.wmqfte.TransferId">414d51205553322e42494e44494e47538b0f404d035c0020</
metaData>
      <metaData key="com.ibm.wmqfte.Priority">0</metaData>
    </metaDataSet>
  </transferSet>
  <statistics>
    <actualStartTime>2011-01-26T13:12:58.634Z</actualStartTime>
    <retryCount>0</retryCount>
    <numFileFailures>0</numFileFailures>
    <numFileWarnings>0</numFileWarnings>
  </statistics>
</transaction>

```

Exemples de messages de journal de transfert de fichier ayant échoué

Lorsqu'un transfert est effectué, les messages sont publiés dans SYSTEM.FTE avec une chaîne de rubrique `Log/nom_agent/ID_transfert`. Les exemples XML montrent les messages de journal pour un transfert de fichier qui échoue au démarrage, en cours et terminé.

Echec du transfert de fichiers-démarré

```
<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  ID="414d51205553322e42494e44494e47538b0f404d03620020"
  agentRole="sourceAgent"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2011-01-26T13:19:15.767Z">started</action>
  <sourceAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
    <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
  </sourceAgent>
  <destinationAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER"/>
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
    <mqmdUserID>mqm</mqmdUserID>
  </originator>
  <transferSet startTime="2011-01-26T13:19:15.767Z" total="1" bytesSent="0">
    <metaDataSet>
      <metaData key="com.ibm.wmqfte.SourceAgent">AGENT_JUPITER</metaData>
      <metaData key="com.ibm.wmqfte.DestinationAgent">AGENT_JUPITER</metaData>
      <metaData key="com.ibm.wmqfte.MqmdUser">mqm</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingUser">mqm</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingHost">example.com.</metaData>
      <metaData key="com.ibm.wmqfte.TransferId">414d51205553322e42494e44494e47538b0f404d03620020</
metaData>
      <metaData key="com.ibm.wmqfte.Priority">0</metaData>
    </metaDataSet>
  </transferSet>
</transaction>
```

Echec du transfert de fichier-progression

```
<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  ID="414d51205553322e42494e44494e47538b0f404d03620020"
  agentRole="sourceAgent"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2011-01-26T13:19:15.944Z">progress</action>
  <sourceAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
    <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
  </sourceAgent>
  <destinationAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
    <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
  </destinationAgent>
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
    <mqmdUserID>mqm</mqmdUserID>
  </originator>
  <transferSet index="0" size="1" startTime="2011-01-26T13:19:15.767Z" total="1" bytesSent="0">
    <item mode="binary">
      <source disposition="leave" type="file">
        <file size="0" last-modified="2011-01-26T13:10:19.000Z">/srv/nfs/incoming/file01.txt</
file>
        <checksum method="MD5">d41d8cd98f00b204e9800998ecf8427e</checksum>
      </source>
      <destination type="file">
        <file>/srv/nfs/outgoing/file01.txt</file>
      </destination>
      <status resultCode="1">
        <supplement>BFGIO0006E: File "/srv/nfs/outgoing/file01.txt" already exists.</
supplement>
      </status>
    </item>
  </transferSet>
</transaction>
```

Echec du transfert de fichier-terminé

```
<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
```

```

        version="4.00"
        ID="414d51205553322e42494e44494e47538b0f404d03620020"
        agentRole="sourceAgent"
        xsi:noNamespaceSchemaLocation="TransferLog.xsd"
        xmlns=""
<action time="2011-01-26T13:19:15.948Z">completed</action>
<sourceAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
  <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
</sourceAgent>
<destinationAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER">
  <systemInfo architecture="x86" name="Linux" version="2.6.31-21-generic"/>
</destinationAgent>
<originator>
  <hostName>example.com.</hostName>
  <userID>mqm</userID>
  <mqmdUserID>mqm</mqmdUserID>
</originator>
<status resultCode="40">
  <supplement>BFGRP0034I: The file transfer request has
    completed with no files being transferred.
  </supplement>
</status>
<transferSet startTime="2011-01-26T13:19:15.767Z" total="1" bytesSent="0">
  <metaDataSet>
    <metaData key="com.ibm.wmqfte.SourceAgent">AGENT_JUPITER</metaData>
    <metaData key="com.ibm.wmqfte.DestinationAgent">AGENT_JUPITER</metaData>
    <metaData key="com.ibm.wmqfte.MqmdUser">mqm</metaData>
    <metaData key="com.ibm.wmqfte.OriginatingUser">mqm</metaData>
    <metaData key="com.ibm.wmqfte.OriginatingHost">example.com.</metaData>
    <metaData key="com.ibm.wmqfte.TransferId">414d51205553322e42494e44494e47538b0f404d03620020</
metaData>
    <metaData key="com.ibm.wmqfte.Priority">0</metaData>
  </metaDataSet>
</transferSet>
<statistics>
  <actualStartTime>2011-01-26T13:19:15.878Z</actualStartTime>
  <retryCount>0</retryCount>
  <numFileFailures>1</numFileFailures>
  <numFileWarnings>0</numFileWarnings>
</statistics>
</transaction>

```

Exemple de message de journal de transfert de fichier déclenché

Lorsqu'un transfert est en cours, des messages sont publiés dans SYSTEM.FTE avec une chaîne de rubrique `Log/nom_agent/ID_transfert`. L'exemple XML montre le message de journal qui est créé lorsqu'un transfert de fichier contenant une condition de déclenchement est démarré.

Réussite du transfert de fichier unique de déclencheur-démarré

```

<?xml version="1.0" encoding="UTF-8"?>
<transaction version="1.00"
  ID="414d5120514d31202020202020202020207e970d492000a102" agentRole="sourceAgent"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2008-11-02T22:05:18.703Z">started</action>
  <sourceAgent agent="FTEAGENT" QMgr="QM1">
    <systemInfo architecture="x86" name="Windows 7"
      version="6.1 build 7601 Service Pack 1"/>
  </sourceAgent>
  <destinationAgent agent="FTEAGENT" QMgr="QM1"/>
  <originator>
    <hostName>reportserver.com</hostName>
    <userID>USER1</userID>
    <mqmdUserID>USER1 </mqmdUserID>
  </originator>
  <trigger log="yes">
    <fileExist comparison="" value="exist">c:\trigger.txt</fileExist>
  </trigger>
  <transferSet startTime="2008-11-02T22:05:18.703Z" total="1"></transferSet>
</transaction>

```


Sortie du transfert de fichier unique-procédure démarrée

```
<?xml version="1.0" encoding="UTF-8"?>
<transaction version="1.00"
  ID="414d5120514d31202020202020202020207e970d492000d502" agentRole="sourceAgent"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2008-11-02T22:36:13.046Z">started</action>
  <sourceAgent agent="FTEAGENT" QMgr="QM1">
    <systemInfo architecture="x86" name="Windows 7"
      version="6.1 build 7601 Service Pack 1"/>
  </sourceAgent>
  <destinationAgent agent="FTEAGENT" QMgr="QM1"/>
  <originator>
    <hostName>reportserver.com</hostName>
    <userID>USER1</userID>
    <mqmdUserID>USER1 </mqmdUserID>
  </originator>
  <transferSet startTime="2008-11-02T22:36:13.046Z" total="1">
    <metaDataSet>
      <metaData key="testkey1">testvalue1</metaData>
      <metaData key="testkey2">testvalue2</metaData>
    </metaDataSet>
  </transferSet>
</transaction>
```

Quitter le transfert de fichier unique-terminé

```
<?xml version="1.0" encoding="UTF-8"?>
<transaction version="1.00"
  ID="414d5120514d31202020202020202020207e970d492000d502"
  agentRole="sourceAgent"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2008-11-02T22:36:13.546Z">completed</action>
  <sourceAgent agent="FTEAGENT" QMgr="QM1">
    <startExits>
      <exit name="class testExits.SourceExit1">
        <status resultCode="proceed">
          <supplement>Source Start, modified metadata</supplement>
        </status>
      </exit>
    </startExits>
    <endExits>
      <exit name="class testExits.SourceExit1">
        <status>
          <supplement>Source End</supplement>
        </status>
      </exit>
    </endExits>
    <systemInfo architecture="x86" name="Windows 7"
      version="6.1 build 7601 Service Pack 1"/>
  </sourceAgent>
  <destinationAgent agent="FTEAGENT" QMgr="QM1">
    <startExits>
      <exit name="class testExits.DestinationExitProceed">
        <status resultCode="proceed">
          <supplement>Destination start, with proceed</supplement>
        </status>
      </exit>
    </startExits>
    <endExits>
      <exit name="class testExits.DestinationExitProceed">
        <status>
          <supplement>destination end</supplement>
        </status>
      </exit>
    </endExits>
    <systemInfo architecture="x86" name="Windows 7"
      version="6.1 build 7601 Service Pack 1"/>
  </destinationAgent>
  <originator>
    <hostName>reportserver.com</hostName>
    <userID>USER1</userID>
    <mqmdUserID>USER1 </mqmdUserID>
  </originator>
```

```

<transferSet startTime="2008-11-02T22:36:13.046Z" total="1">
  <metaDataSet>
    <metaData key="newkey2">newvalue2</metaData>
    <metaData key="newkey1">newvalue1</metaData>
    <metaData key="newkey4">newvalue4</metaData>
    <metaData key="newkey3">newvalue3</metaData>
    <metaData key="newkey5">newvalue5</metaData>
    <metaData key="testkey1">testvalue1</metaData>
    <metaData key="testkey2">testvalue2</metaData>
  </metaDataSet>
</transferSet>
</transaction>

<!--
  In this example the source transfer start exit has modified the
  metadata as follows:

  Added keys and values for:
  newkey1, newvalue1
  newkey2, newvalue2
  newkey3, newvalue3
  newkey4, newvalue4
  newkey5, newvalue5

  Replaced values for:
  key1 to modifiedValue1

  Deleted keys and values for:
  key2
-->

```

Quittez l'annulation du transfert de fichier unique-annulé

```

<?xml version="1.0" encoding="UTF-8"?>
<transaction version="1.00"
  ID="414d5120514d312020202020202020207e970d492000c702" agentRole="sourceAgent"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2008-11-02T22:25:59.328Z">cancelled</action>
  <sourceAgent agent="FTEAGENT" QMgr="QM1">
    <startExits>
      <exit name="class testExits.SourceExit1">
        <status resultCode="proceed">
          <supplement>Source Start, modified metadata</supplement>
        </status>
      </exit>
    </startExits>
    <endExits>
      <exit name="class testExits.SourceExit1">
        <status>
          <supplement>Source End</supplement>
        </status>
      </exit>
    </endExits>
    <systemInfo architecture="x86" name="Windows 7"
      version="6.1 build 7601 Service Pack 1"/>
  </sourceAgent>
  <destinationAgent agent="FTEAGENT" QMgr="QM1">
    <startExits>
      <exit name="class testExits.DestinationExit1">
        <status resultCode="cancelTransfer">
          <supplement>Destination start, with cancel</supplement>
        </status>
      </exit>
    </startExits>
    <endExits>
      <exit name="class testExits.DestinationExit1">
        <status>
          <supplement>destination end</supplement>
        </status>
      </exit>
    </endExits>
    <systemInfo architecture="x86" name="Windows 7"
      version="6.1 build 7601 Service Pack 1"/>
  </destinationAgent>
  <originator>
    <hostName>reportserver.com</hostName>
    <userID>USER1</userID>
  </originator>

```

```

    <mqmdUserID>USER1          </mqmdUserID>
  </originator>
  <transferSet startTime="2008-11-02T22:25:59.078Z" total="1"/>
</transaction>

```

Exemples de message de journal de transfert de pont Connect:Direct

L'élément destinationAgent ou sourceAgent contient des attributs supplémentaires lorsque l'agent cible ou l'agent source est un agent de pont Connect:Direct . Le message de journal Démarré ne contient qu'un sous-ensemble des informations sur le transfert Connect:Direct . Les messages de journal Progression et Terminé contiennent des informations complètes sur le transfert Connect:Direct .

L'agent source est l'agent de pont Connect:Direct

Démarré :

```

<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  ID="414d5120514d5f696b6b796f20202020a704654d20092507"
  agentRole="sourceAgent"
  version="4.00"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2011-03-07T13:05:01.838Z">started</action>
  <sourceAgent QMgr="QM_KUIPER" agent="VARUNA" agentType="CD_BRIDGE" bridgeNode="CDNODE_VARUNA">
    <systemInfo architecture="x86" name="Windows 7" version="6.1 build 7601 Service Pack 1"/>
  </sourceAgent>
  <destinationAgent QMgr="QM_KUIPER" agent="IXION"/>
  <originator>
    <hostName>kuiper.example.com.</hostName>
    <userID>sol</userID>
    <mqmdUserID>sol</mqmdUserID>
  </originator>
  <transferSet bytesSent="0" startTime="2011-03-07T13:05:01.838Z" total="1">
    <metaDataSet>
      <metaData key="com.ibm.wmqfte.SourceAgent">VARUNA</metaData>
      <metaData key="com.ibm.wmqfte.DestinationAgent">IXION</metaData>
      <metaData key="com.ibm.wmqfte.MqmdUser">sol</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingUser">sol</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingHost">kuiper.example.com.</metaData>
      <metaData key="com.ibm.wmqfte.TransferId">414d5120514d5f696b6b796f20202020a704654d20092507</
metaData>
      <metaData key="com.ibm.wmqfte.Priority">0</metaData>
    </metaDataSet>
  </transferSet>
</transaction>

```

Progression :

```

<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  ID="414d5120514d5f696b6b796f20202020a704654d20092507"
  agentRole="sourceAgent"
  version="4.00"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2011-03-07T13:05:03.448Z">progress</action>
  <sourceAgent QMgr="QM_KUIPER" agent="VARUNA" agentType="CD_BRIDGE"
    bridgeNode="CDNODE_VARUNA" pnode="CDNODE_VARUNA" snode="CDNODE_ERIS">
    <systemInfo architecture="x86" name="Windows 7" version="6.1 build 7601 Service Pack 1"/>
  </sourceAgent>
  <destinationAgent QMgr="QM_KUIPER" agent="IXION" agentType="STANDARD">
    <systemInfo architecture="x86" name="Windows 7" version="6.1 build 7601 Service Pack 1"/>
  </destinationAgent>
  <originator>
    <hostName>kuiper.example.com.</hostName>
    <userID>sol</userID>
    <mqmdUserID>sol</mqmdUserID>
  </originator>
  <transferSet bytesSent="48" index="0" size="1" startTime="2011-03-07T13:05:01.838Z" total="1">
    <item mode="binary">
      <source disposition="leave" processName="f2007567" processNumber="68" type="file">
        <file last-modified="2011-03-07T13:05:02.573Z" size="4">CDNODE_ERIS:D:/AGENTS/
CDNODE_ERIS/test.txt</file>
        <checksum method="MD5">098f6bcd4621d373cade4e832627b4f6</checksum>
      </source>
      <destination type="file">

```

```

        <file last-modified="2011-03-07T13:05:03.338Z" size="4">D:\AGENTS\IXION\test.txt</file>
        <checksum method="MD5">098f6bcd4621d373cade4e832627b4f6</checksum>
      </destination>
      <status resultCode="0"/>
    </item>
  </transferSet>
</transaction>

```

Opération terminée :

```

<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  ID="414d5120514d5f696b6b796f20202020a704654d20092507"
  agentRole="sourceAgent"
  version="4.00" xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2011-03-07T13:05:03.495Z">completed</action>
  <sourceAgent QMgr="QM_KUIPER" agent="VARUNA" agentType="CD_BRIDGE"
    bridgeNode="CDNODE_VARUNA" pnode="CDNODE_VARUNA" snode="CDNODE_ERIS">
    <systemInfo architecture="x86" name="Windows 7" version="6.1 build 7601 Service Pack 1"/>
  </sourceAgent>
  <destinationAgent QMgr="QM_KUIPER" agent="IXION" agentType="STANDARD">
    <systemInfo architecture="x86" name="Windows 7" version="6.1 build 7601 Service Pack 1"/>
  </destinationAgent>
  <originator>
    <hostName>kuiper.example.com.</hostName>
    <userID>sol</userID>
    <mqmdUserID>sol</mqmdUserID>
  </originator>
  <status resultCode="0">
    <supplement>BFGRP0032I: The file transfer request has successfully completed.</supplement>
  </status>
  <transferSet bytesSent="48" startTime="2011-03-07T13:05:01.838Z" total="1">
    <metaDataSet>
      <metaData key="com.ibm.wmqfte.SourceAgent">VARUNA</metaData>
      <metaData key="com.ibm.wmqfte.DestinationAgent">IXION</metaData>
      <metaData key="com.ibm.wmqfte.MqmdUser">sol</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingUser">sol</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingHost">kuiper.example.com.</metaData>
      <metaData key="com.ibm.wmqfte.TransferId">414d5120514d5f696b6b796f20202020a704654d20092507</
metaData>
      <metaData key="com.ibm.wmqfte.Priority">0</metaData>
    </metaDataSet>
  </transferSet>
  <statistics>
    <actualStartTime>2011-03-07T13:05:02.041Z</actualStartTime>
    <retryCount>0</retryCount>
    <numFileFailures>0</numFileFailures>
    <numFileWarnings>0</numFileWarnings>
  </statistics>
</transaction>

```

L'agent de destination est l'agent de pont Connect:Direct

Démarré :

```

<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  ID="414d5120514d5f696b6b796f20202020a704654d2008e102"
  agentRole="sourceAgent"
  version="4.00"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2011-03-07T10:29:44.854Z">started</action>
  <sourceAgent QMgr="QM_ASTEROID" agent="PALLAS" agentType="STANDARD">
    <systemInfo architecture="x86" name="Windows 7" version="6.1 build 7601 Service Pack 1"/>
  </sourceAgent>
  <destinationAgent QMgr="QM_ASTEROID" agent="VESTA"/>
  <originator>
    <hostName>belt.example.com.</hostName>
    <userID>sol</userID>
    <mqmdUserID>sol</mqmdUserID>
  </originator>
  <transferSet bytesSent="0" startTime="2011-03-07T10:29:44.854Z" total="1">
    <metaDataSet>
      <metaData key="com.ibm.wmqfte.SourceAgent">PALLAS</metaData>
      <metaData key="com.ibm.wmqfte.DestinationAgent">VESTA</metaData>
      <metaData key="com.ibm.wmqfte.MqmdUser">sol</metaData>
      <metaData key="com.ibm.wmqfte.OriginatingUser">sol</metaData>

```

```

        <metaData key="com.ibm.wmqfte.OriginatingHost">belt.example.com.</metaData>
        <metaData key="com.ibm.wmqfte.TransferId">414d5120514d5f696b6b796f20202020a704654d2008e102</
metaData>
        <metaData key="com.ibm.wmqfte.Priority">0</metaData>
    </metaDataSet>
</transferSet>
</transaction>

```

Progression :

```

<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
    ID="414d5120514d5f696b6b796f20202020a704654d2008e102"
    agentRole="sourceAgent"
    version="4.00"
    xsi:noNamespaceSchemaLocation="TransferLog.xsd"
    xmlns="">
    <action time="2011-03-07T10:29:46.682Z">progress</action>
    <sourceAgent QMgr="QM_asteroid" agent="PALLAS" agentType="STANDARD">
        <systemInfo architecture="x86" name="Windows 7" version="6.1 build 7601 Service Pack 1"/>
    </sourceAgent>
    <destinationAgent QMgr="QM_asteroid" agent="VESTA" agentType="CD_BRIDGE"
        bridgeNode="CDNODE_VESTA" pNode="CDNODE_VESTA" sNode="CDNODE_HYGIEA">
        <systemInfo architecture="x86" name="Windows 7" version="6.1 build 7601 Service Pack 1"/>
    </destinationAgent>
    <originator>
        <hostName>belt.example.com</hostName>
        <userID>sol</userID>
        <mqmdUserID>sol</mqmdUserID>
    </originator>
    <transferSet bytesSent="48" index="0" size="1" startTime="2011-03-07T10:29:44.854Z" total="1">
        <item mode="binary">
            <source disposition="leave" type="file">
                <file last-modified="2011-03-04T14:53:28.323Z" size="4">D:\AGENTS\PALLAS\test.txt</
file>
                <checksum method="MD5">098f6bcd4621d373cade4e832627b4f6</checksum>
            </source>
            <destination processName="f2006965" processNumber="59" type="file">
                <file size="4">CDNODE_VESTA:D:/AGENTS/CDNODE_VESTA/test.txt</file>
                <checksum method="MD5">098f6bcd4621d373cade4e832627b4f6</checksum>
            </destination>
            <status resultCode="0"/>
        </item>
    </transferSet>
</transaction>

```

Opération terminée :

```

<?xml version="1.0" encoding="UTF-8"?>
<transaction xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
    ID="414d5120514d5f696b6b796f20202020a704654d2008e102"
    agentRole="sourceAgent"
    version="4.00"
    xsi:noNamespaceSchemaLocation="TransferLog.xsd"
    xmlns="">
    <action time="2011-03-07T10:29:46.698Z">completed</action>
    <sourceAgent QMgr="QM_asteroid" agent="PALLAS" agentType="STANDARD">
        <systemInfo architecture="x86" name="Windows 7" version="6.1 build 7601 Service Pack 1"/>
    </sourceAgent>
    <destinationAgent QMgr="QM_asteroid" agent="VESTA" agentType="CD_BRIDGE"
        bridgeNode="CDNODE_VESTA" pNode="CDNODE_VESTA" sNode="CDNODE_HYGIEA">
        <systemInfo architecture="x86" name="Windows 7" version="6.1 build 7601 Service Pack 1"/>
    </destinationAgent>
    <originator>
        <hostName>belt.example.com</hostName>
        <userID>sol</userID>
        <mqmdUserID>sol</mqmdUserID>
    </originator>
    <status resultCode="0">
        <supplement>BFGRP0032I: The file transfer request has successfully completed.</supplement>
    </status>
    <transferSet bytesSent="48" startTime="2011-03-07T10:29:44.854Z" total="1">
        <metaDataSet>
            <metaData key="com.ibm.wmqfte.SourceAgent">PALLAS</metaData>
            <metaData key="com.ibm.wmqfte.DestinationAgent">VESTA</metaData>
            <metaData key="com.ibm.wmqfte.MqmdUser">sol</metaData>
            <metaData key="com.ibm.wmqfte.OriginatingUser">sol</metaData>
            <metaData key="com.ibm.wmqfte.OriginatingHost">belt.example.com.</metaData>
            <metaData key="com.ibm.wmqfte.TransferId">414d5120514d5f696b6b796f20202020a704654d2008e102</
metaData>

```

```

        <metaData key="com.ibm.wmqfte.Priority">0</metaData>
    </metaDataSet>
</transferSet>
<statistics>
    <actualStartTime>2011-03-07T10:29:45.010Z</actualStartTime>
    <retryCount>0</retryCount>
    <numFileFailures>0</numFileFailures>
    <numFileWarnings>0</numFileWarnings>
</statistics>
</transaction>

```

Formats de message du journal de transfert de fichiers planifiés

Chaque fois qu'une demande de transfert de fichier planifié est traitée par l'agent, un message de journal de planification est publié dans le gestionnaire de files d'attente de coordination (sur son système SYSTEM.FTE/Log/nom de l'agent/ID planification). Ce message est conforme au schéma XML ScheduleLog.xsd.

Schéma

Le schéma suivant décrit les éléments valides dans un message XML de journal de planification.

```

<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema">
  <xsd:include schemaLocation="fteutils.xsd"/>

  <xsd:element name="schedulelog">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="originator" type="hostUserIDType"
          maxOccurs="1" minOccurs="0"/>
        <xsd:element name="action" type="actionType"
          maxOccurs="1" minOccurs="1"/>
        <xsd:element name="schedule" type="scheduleType"
          maxOccurs="1" minOccurs="0"/>
        <xsd:element name="sourceAgent" type="agentType"
          maxOccurs="1" minOccurs="0"/>
        <xsd:element name="destinationAgent" type="agentClientType"
          maxOccurs="1" minOccurs="0"/>
        <xsd:element name="status" type="statusType"
          maxOccurs="1" minOccurs="0"/>
        <xsd:element name="transferSet" type="transferSetType"
          maxOccurs="1" minOccurs="0"/>
        <xsd:element name="job" type="jobType"
          maxOccurs="1" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="version" type="versionType" use="required"/>
      <xsd:attribute name="ID" type="xsd:string" use="required"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="actionType">
    <xsd:simpleContent>
      <xsd:extension base="actionEnumType">
        <xsd:attribute name="time" type="xsd:dateTime" use="required"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>

  <xsd:simpleType name="actionEnumType">
    <xsd:restriction base="xsd:token">
      <xsd:enumeration value="submit"/>
      <xsd:enumeration value="delete"/>
      <xsd:enumeration value="expire"/>
      <xsd:enumeration value="skipped"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name="transferSetType">
    <xsd:sequence>
      <xsd:element name="item" type="itemType"
        maxOccurs="unbounded" minOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute name="size" type="xsd:int" use="required"/>
    <xsd:attribute name="priority" type="priorityType" use="optional"/>
  </xsd:complexType>

```

```

<xsd:complexType name="itemType">
  <xsd:sequence>
    <xsd:element name="source" type="fileSourceType"
      maxOccurs="1" minOccurs="1"/>
    <xsd:element name="destination" type="fileDestinationType"
      maxOccurs="1" minOccurs="1"/>
  </xsd:sequence>
  <xsd:attribute name="mode" type="modeType" use="required"/>
  <xsd:attribute name="checksumMethod" type="checksumMethod" use="required"/>
</xsd:complexType>
</xsd:schema>

```

Description du message de journal de planification

Les éléments et les attributs utilisés dans le message de journal de planification sont décrits ci-dessous:

< planning >

Élément de groupe qui décrit un seul transfert de fichier planifié soumis.

Attribut	Description
version	Indique la version de cet élément, telle que détaillée par Managed File Transfer.
ID	Identificateur unique du transfert de fichiers de planification soumis.

< originaire >

Élément de groupe qui contient les éléments spécifiant l'émetteur de la demande.

< hostName >

Nom d'hôte du système sur lequel se trouve le fichier source.

< userID >

ID utilisateur à l'origine du transfert de fichiers.

< IDmqmdUser >

ID utilisateur MQMD fourni dans le descripteur de message (MQMD)

< action >

Indique l'action à effectuer avec le transfert planifié correspondant à l'attribut ID de l'élément < schedulelog >. Cet élément doit avoir l'une des valeurs suivantes:

- submit-nouveau transfert planifié
- delete-annuler le transfert de planification
- expire-poste de transfert de planning sur le point d'être traité
- ignoré-un transfert planifié ne peut pas être démarré car l'agent est hors ligne. Ce message est consigné lorsque l'agent devient disponible pour indiquer que le transfert a été ignoré.

Attribut	Description
heure	Indique la date et l'heure de publication de l'entrée de journal (au format date / heure).

< sourceAgent >

Indique le nom de l'agent sur le système où se trouve le fichier source.

Attribut	Description
Agent	Indique le nom de l'agent.
QMgr	Nom du gestionnaire de files d'attente d'agent.

< destinationAgent >

Indique le nom de l'agent sur le système vers lequel vous souhaitez transférer le fichier.

Attribut	Description
Agent	Indique le nom de l'agent.
QMgr	Nom du gestionnaire de files d'attente d'agent.

< statut >

Code de résultat et messages complémentaires.

<transferSet >

Indique un groupe de transferts de fichiers que vous souhaitez effectuer ensemble. Lors de la transmission, <transferSet> est un élément de groupe contenant des éléments < item >.

Attribut	Description
taille	Indique le nombre d'éléments de transfert.
priorité	Niveau de priorité du transfert. La priorité est une valeur comprise entre 0 et 9, où 0 est la priorité la plus basse. Le niveau de priorité par défaut est 0 et, par défaut, le transfert utilise le niveau de priorité de l'agent source.

<item >

Élément de groupe qui contient des éléments spécifiant les noms et les emplacements des fichiers source et de destination.

Attribut	Description
mode	Indique que le mode de transfert est binaire ou texte.
checksumMethod	Indique le type d'algorithme de hachage qui génère le résumé du message pour créer la signature numérique. Les valeurs admises sont MD5 ou aucune

< source >

Élément de groupe qui contient les éléments < file > et < checksum > pour le fichier sur le système source.

Attribut	Description
recursive	Indique que les fichiers sont transférés de manière récursive dans des sous-répertoires lorsque l'élément < source > est un répertoire ou contient des caractères génériques.
disposition	Indique l'action effectuée sur l'élément < source > lorsque < source > a été transféré vers sa destination. Les options valides sont les suivantes: <ul style="list-style-type: none"> • leave-les fichiers source restent inchangés. • delete-les fichiers source sont supprimés du système source après le transfert du fichier source.

< destination >

Élément de groupe contenant les éléments < file > et < checksum > du fichier sur le système de destination.

Attribut	Description
type	Type de fichier ou de répertoire sur la destination. Les options valides sont les suivantes: <ul style="list-style-type: none"> • file-spécifie un fichier comme destination • directory-spécifie un répertoire comme destination •  dataset-spécifie un fichier z/OS comme destination

Attribut	Description
	<ul style="list-style-type: none"> •  PDS-spécifie un fichier partitionné z/OS comme destination
existe	<p>Indique l'action effectuée si un fichier de destination existe sur le système de destination. Les options valides sont les suivantes:</p> <ul style="list-style-type: none"> • error-signale une erreur et le fichier n'est pas transféré. • overwrite-écrase le fichier de destination existant.

<fichier>

Indique le nom du fichier à transférer. Utilisez le chemin qualifié complet dans le format correspondant à votre système d'exploitation, par exemple C : /from/here . txt. N'utilisez pas d'URI de fichier.

Attribut	Description
codage	Codage d'un transfert de fichier texte.
EOL	<p>Indique le marqueur de fin de ligne. Les valeurs admises sont les suivantes:</p> <ul style="list-style-type: none"> • Saut de ligne-caractère de saut de ligne uniquement • CRLF-retour chariot et séquence de caractères de saut de ligne

< travail >

Élément de groupe qui contient un élément spécifiant les détails du travail. < job > est un identificateur de nom de travail défini par l'utilisateur qui est ajouté au message du journal lorsque le transfert a commencé. Cet élément < job > est identique à l'élément < job > inclus dans le message de demande de transfert, décrit dans la rubrique suivante: [«Format de message de demande de transfert de fichier»](#), à la page 804.

< nom >

La valeur du nom peut être n'importe quelle chaîne.

Exemples

Des exemples de messages XML conformes à ce schéma sont fournis pour chacune des actions de transfert planifiées suivantes:

- [Un transfert planifié est créé](#)
- [Un transfert planifié est annulé](#)
- [Un transfert planifié expire](#)

Les transferts démarrés par une planification sont consignés de la même manière qu'un transfert standard. Pour obtenir des exemples de messages de journal pour les transferts démarrés par une planification, voir [«Exemples de message de journal de transfert de fichier planifié»](#), à la page 787.

Exemples de message de journal de transfert de fichiers de planification

Exemples de messages publiés dans SYSTEM.FTE avec une chaîne de rubrique Log/nom_agent/schedule_ID lorsqu'une action de transfert planifiée se produit.

Message du journal de transfert planifié

Chaque fois qu'une demande de transfert de fichier planifié est traitée par l'agent, un message de journal de planification est publié dans le gestionnaire de files d'attente de coordination (sur son

système SYSTEM.FTE/Log/nom de l'agent/ID planification). Ce message est conforme au schéma XML ScheduleLog.xsd . Exemple :

```
<?xml version="1.0" encoding="UTF-8"?>
<schedulelog version="1.00" ID="5"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ScheduleLog.xsd">
  <originator>
    <hostName>reportserver.com</hostName>
    <userID>USER1</userID>
  </originator>
  <action time="2008-11-23T21:32:01Z">submit</action>
  <schedule>
    <submit timebase="admin" timezone="Europe/London">2008-11-23T22:00</submit>
  </schedule>
  <sourceAgent agent="FTEAGENT" QMgr="QM1" />
  <destinationAgent agent="FTEAGENT" QMgr="QM1" />
  <status resultCode="0" />
  <transferSet size="1" priority="0">
    <item mode="binary" checksumMethod="MD5">
      <source recursive="false" disposition="leave">
        <file>c:\sourcefiles\source1.doc</file>
      </source>
      <destination type="file" exist="overwrite">
        <file>c:\destinationfiles\dest1.doc</file>
      </destination>
    </item>
  </transferSet>
</schedulelog>
```

Ce message est un journal contenant les informations suivantes:

- A l'origine de la demande
- Date de soumission de la demande
- Date de début du transfert planifié
- Détails de l'agent source et de destination
- Spécification de transfert

L'attribut ID de l'élément < schedulelog> est un ID unique pour ce transfert planifié (dans l'agent source). Cet ID est utilisé pour corrélérer les entrées de planification avec les transferts de fichiers réels.

La valeur de l'élément < action> de l'élément submit confirme que la demande a été reçue.

Message de journal d'annulation de transfert planifié

Lorsqu'une demande d'annulation d'un transfert de fichier planifié en attente est reçue par l'agent, le message suivant est publié dans SYSTEM.FTE/Log/nom_agent :

```
<?xml version="1.0" encoding="UTF-8"?>
<schedulelog version="1.00" ID="5"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ScheduleLog.xsd">
  <originator>
    <hostName>reportserver.com</hostName>
    <userID>USER1</userID>
  </originator>
  <action time="2008-11-23T21:56:27Z">delete</action>
  <status resultCode="0" />
</schedulelog>
```

La valeur de l'attribut ID correspond à l'ID de l'ID de demande de transfert en attente dans le message de planification.

Message de journal d'expiration de transfert planifié

Lorsque l'heure en cours correspond à l'heure du premier transfert de fichier en attente dans la liste des planifications (comme indiqué par la valeur de l'élément < next>), un message de journal de planification est publié pour indiquer que l'entrée de transfert planifié a expiré:

```
<?xml version="1.0" encoding="UTF-8"?>
<schedulelog xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00" ID="3"
  xsi:noNamespaceSchemaLocation="ScheduleLog.xsd">
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
    <mqmdUserID>mqm</mqmdUserID>
  </originator>
  <action time="2011-01-26T13:03:26Z">expire</action>
  <sourceAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER"/>
  <status resultCode="0"/>
</schedulelog>
```

La valeur de l'élément < action> "expire" confirme que l'entrée de planification a été supprimée de la liste des planifications et qu'elle est en cours de traitement. Un message de planification pour l'agent est publié avec l'entrée arrivée à expiration qui n'est plus présente.

Référence associée

«Formats de message du journal de transfert de fichiers planifiés», à la page 793

Chaque fois qu'une demande de transfert de fichier planifié est traitée par l'agent, un message de journal de planification est publié dans le gestionnaire de files d'attente de coordination (sur son système SYSTEM.FTE/Log/nom de l'agent/ID planification). Ce message est conforme au schéma XML ScheduleLog.xsd .

«Exemples de message de journal de transfert de fichier planifié», à la page 787

Lorsqu'un transfert est en cours, des messages sont publiés dans SYSTEM.FTE avec une chaîne de rubrique Log/nom_agent/ID_transfert. Les exemples XML montrent les messages de journal qui sont créés lorsqu'un transfert de fichier est effectué suite à une planification.

Format des messages du journal du moniteur MFT

Les messages de journal du moniteur sont publiés dans SYSTEM.FTE avec une chaîne de rubrique Log/agent_name/Monitors/monitor_name/monitor_ID.

Si vous souhaitez collecter des données ou afficher des actions de moniteur, configurez un abonnement à une rubrique générique adaptée aux moniteurs qui vous intéressent. Exemple :

```
Log/#
```

ou,

```
Log/agent_name/#
```

Cet abonnement peut être durable ou non durable. Les abonnements durables continuent d'exister lorsque la connexion d'une application d'abonnement au gestionnaire de files d'attente est fermée. Les abonnements non durables existent uniquement tant que la connexion d'une application d'abonnement au gestionnaire de files d'attente reste ouverte.

Le document de schéma MonitorLog.xsd se trouve dans le répertoire MQ_INSTALLATION_PATH/mqft/samples/schema . Le MonitorLog.xsd schéma importe fteutils.xsd, qui se trouve dans le même répertoire.

Schéma

Le schéma suivant décrit les éléments qui sont valides dans un message XML de journal de surveillance.

```
<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema">
  <xsd:include schemaLocation="fteutils.xsd"/>
  <xsd:element name="monitorLog">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="originator" type="hostUserIDType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="references" type="referencesType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="action" type="monitorActionType" maxOccurs="1"
minOccurs="1"/>
        <xsd:element name="monitorAgent" type="agentType" maxOccurs="1"
minOccurs="1"/>
        <xsd:element name="status" type="statusType" maxOccurs="1"
minOccurs="1"/>
        <xsd:element name="monitorMetaData" type="monitorMetaDataType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="monitorExits" type="exitGroupType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="jobDetails" type="jobType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="taskXMLRequest" type="taskXMLRequestType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="monitorXMLRequest" type="monitorXMLRequestType"
maxOccurs="1" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="version" type="versionType" use="required"/>
      <xsd:attribute name="monitorName" type="xsd:string" use="required"/>
      <xsd:attribute name="referenceId" type="xsd:string" use="optional"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="monitorActionType">
    <xsd:simpleContent>
      <xsd:extension base="monitorActionEnumType">
        <xsd:attribute name="time" type="xsd:dateTime" use="required"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>

  <xsd:simpleType name="monitorActionEnumType">
    <xsd:restriction base="xsd:token">
      <xsd:enumeration value="create"/>
      <xsd:enumeration value="delete"/>
      <xsd:enumeration value="start"/>
      <xsd:enumeration value="stop"/>
      <xsd:enumeration value="triggerSatisfied"/>
      <xsd:enumeration value="triggerNotSatisfied"/>
      <xsd:enumeration value="triggerFail"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name="monitorMetaDataType">
    <xsd:sequence>
      <xsd:element name="originalMetaData" type="metaDataSetType" maxOccurs="1" minOccurs="0"/>
      <xsd:element name="updatedMetaData" type="metaDataSetType" maxOccurs="unbounded"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="taskXMLRequestType">
    <xsd:sequence>
      <xsd:element name="originalRequest" type="xsd:string" maxOccurs="1" minOccurs="1"/>
      <xsd:element name="updatedRequest" type="xsd:string" maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="taskId" type="xsd:string" use="required"/>
  </xsd:complexType>

  <xsd:complexType name="referencesType">
    <xsd:sequence>
      <xsd:element name="createRequest" type="xsd:string" maxOccurs="1" minOccurs="0"/>
      <xsd:element name="taskRequest" type="xsd:string" maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

```

<xsd:complexType name="monitorXMLRequestType">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute name="type" type="xmlContentEnumType" use="required"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>

<xsd:simpleType name="xmlContentEnumType">
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="escapedXML"/>
  </xsd:restriction>
</xsd:simpleType>

</xsd:schema>

```

Description du message de journal du moniteur

Les éléments et les attributs utilisés dans les messages du journal du moniteur sont décrits dans la liste suivante:

<monitorLog>

L'élément de groupe contenant les éléments décrit une action qui a été effectuée par un moniteur.

Attribut	Description
version	Obligatoire. Version du format de message de la liste de surveillance.
monitorName	Obligatoire. Nom du moniteur. Unique pour l'agent sur lequel le moniteur est défini.
referenceId	ID de l'action de surveillance.

<originaire>

Élément de groupe qui contient les éléments spécifiant l'émetteur de la demande.

<hostName>

Nom d'hôte du système sur lequel se trouve le fichier source.

<userID>

ID utilisateur à l'origine du transfert de fichiers.

<IDmqmdUser>

Facultatif. ID utilisateur IBM MQ fourni dans le descripteur de message (MQMD).

<references>

Références aux ID des autres messages associés à cette action de surveillance.

<createRequest>

ID du message de demande XML utilisé pour créer le moniteur.

<taskRequest>

ID du message de demande XML que le moniteur soumet à la suite de cette action.

<action>

Action qui s'est produite, à laquelle ce message de journal est associé. La valeur contenue dans l'élément peut être l'une des suivantes: create, delete, start, stop, triggerSatisfied, triggerNotSatisfied ou triggerFail.

<monitorAgent>

Agent qui surveille la ressource.

Attribut	Description
Agent	Obligatoire. Nom de l'agent.
QMgr	Facultatif. Nom du gestionnaire de files d'attente auquel l'agent se connecte.
bridgeURL	Facultatif. Si l'agent est un agent de pont de protocole, URL du serveur de protocole.

< statut >

Statut de l'action du moniteur de ressources en cours de consignation.

Attribut	Description
resultCode	Obligatoire. Code de résultat de type entier issu de l'action.

< supplément >

Informations supplémentaires sur le statut de l'action du moniteur de ressources en cours de consignation.

< donnéesmonitorMeta >

Élément de groupe contenant les éléments <originalMetaData > et <updatedMetaData >.

< DonnéesoriginalMeta >

Élément contenant un ou plusieurs éléments < metadata > qui décrivent les métadonnées du moniteur avant que l'action ne se produise.

< DonnéesupdatedMeta >

Élément contenant un ou plusieurs éléments < metadata > qui décrivent les métadonnées du moniteur après l'exécution de l'action.

< métadonnées >

Définit une paire clé-valeur de métadonnées. La clé est un attribut de l'élément ; la valeur est le contenu de l'élément.

Attribut	Description
clé	Clé des métadonnées.

< monitorExits >

Élément de groupe contenant un ou plusieurs éléments < exit >.

< expirations >

Élément décrivant un exit exécuté par le moniteur de ressources.

Attribut	Description
nom	Obligatoire. Nom de l'exit du moniteur de ressources.

< statut >

Statut de l'exit du moniteur de ressources en cours de consignation.

Attribut	Description
resultCode	Obligatoire. Code de résultat entier de l'exit.

< supplément >

Informations supplémentaires sur l'état de l'exit du moniteur de ressources en cours de consignation.

< jobDetails >

Élément contenant un seul élément < name >.

< nom >

Nom du travail.

< taskXMLRequest >

Élément de groupe contenant les éléments <originalRequest > et <updatedRequest >.

Attribut	Description
taskId	ID du message de demande de tâche.

<originalRequest>

Élément contenant le message de demande XML échappé pour la tâche exécutée par le moniteur.

<updatedRequest>

Élément contenant le message de demande XML d'échappement mis à jour pour la tâche effectuée par le moniteur.

<monitorXMLRequest>

Demande XML du moniteur.

Attribut	Description
type	Obligatoire. Format des données de demande XML du moniteur dans l'élément <monitorXMLRequest>. La seule valeur valide est escapedXML.

Exemples

Des exemples de messages XML conformes à ce schéma sont fournis pour chacune des actions de surveillance suivantes:

- [Un moniteur est créé](#)
- [La condition d'un moniteur est satisfaite lorsque le moniteur interroge la ressource](#)
- [La condition d'un moniteur n'est pas satisfaite lorsque le moniteur interroge la ressource](#)
- [Un moniteur est supprimé](#)

Référence associée

«Exemples de message de journal de surveillance MFT», à la page 802

Exemples de messages publiés dans SYSTEM.FTE avec une chaîne de rubrique Log/agent_name/monitor_ID lorsqu'une action de surveillance est effectuée.

Exemples de message de journal de surveillance MFT

Exemples de messages publiés dans SYSTEM.FTE avec une chaîne de rubrique Log/agent_name/monitor_ID lorsqu'une action de surveillance est effectuée.

Message de journal créé par le moniteur

```
<?xml version="1.0" encoding="UTF-8"?>
<monitorLog xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  monitorName="MONITORTWO"
  referenceId="414d51205553322e42494e444494e47538b0f404d04410020"
  xsi:noNamespaceSchemaLocation="MonitorLog.xsd">
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
    <mqmdUserID>mqm</mqmdUserID>
  </originator>
  <references>
    <createRequest>414d51205553322e42494e444494e47538b0f404d04410020</createRequest>
  </references>
  <action time="2011-01-26T12:41:24Z">start</action>
  <monitorAgent agent="AGENT_JUPITER" QMgr="QM_JUPITER"/>
  <status resultCode="0"/>
</monitorLog>
```

Message du journal des conditions de surveillance satisfaites

```
<?xml version="1.0" encoding="UTF-8"?>
<monitorLog xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  monitorName="MONITORONE"
  referenceId="414d51205553322e42494e444494e47538b0f404d09430020"
  xsi:noNamespaceSchemaLocation="MonitorLog.xsd">
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
    <mqmdUserID>mqm</mqmdUserID>
```

```

</originator>
<references>
  <createRequest>414d51205553322e42494e44494e47538b0f404d09430020</createRequest>
</references>
<action time="2011-01-26T12:56:46Z">triggerSatisfied</action>
<monitorAgent agent="US2.BINDINGS.FILE" QMgr="US2.BINDINGS" />
<status resultCode="0"/>
<monitorMetaData>
  <originalMetaData>
    <metaData key="AGENTNAME">AGENT_JUPITER</metaData>
    <metaData key="LASTMODIFIEDDATEUTC">2011-01-26</metaData>
    <metaData key="CURRENTTIMESTAMPUTC">20110126125646793</metaData>
    <metaData key="CURRENTTIMESTAMP">20110126125646793</metaData>
    <metaData key="LASTMODIFIEDDATE">2011-01-26</metaData>
    <metaData key="FILENAME">new.completed</metaData>
    <metaData key="LASTMODIFIEDTIMEUTC">12.56</metaData>
    <metaData key="LASTMODIFIEDTIME">12.56</metaData>
    <metaData key="FILESIZE">0</metaData>
    <metaData key="FILEPATH">/srv/nfs/incoming/new.completed</metaData>
  </originalMetaData>
  <updatedMetaData>
    <metaData key="AGENTNAME">AGENT_JUPITER</metaData>
    <metaData key="LASTMODIFIEDDATEUTC">2011-01-26</metaData>
    <metaData key="CURRENTTIMESTAMPUTC">20110126125646793</metaData>
    <metaData key="CURRENTTIMESTAMP">20110126125646793</metaData>
    <metaData key="LASTMODIFIEDDATE">2011-01-26</metaData>
    <metaData key="FILENAME">new.completed</metaData>
    <metaData key="LASTMODIFIEDTIMEUTC">12.56</metaData>
    <metaData key="LASTMODIFIEDTIME">12.56</metaData>
    <metaData key="FILESIZE">0</metaData>
    <metaData key="FILEPATH">/srv/nfs/incoming/new.completed</metaData>
  </updatedMetaData>
</monitorMetaData>
<taskXMLRequest taskId="null">
  <originalRequest>&lt;?xml version="1.0" encoding="UTF-8"?&gt;&lt;request
    xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance" version="4.00"
    xsi:noNamespaceSchemaLocation="FileTransfer.xsd"&gt;&lt;managedTransfer&gt;
      &lt;originator&gt;&lt;hostName&gt;example.com.&lt;/hostName&gt;
      &lt;userID&gt;mqm&lt;/userID&gt;&lt;/originator&gt;
      &lt;sourceAgent QMgr="QM_JUPITER" agent="AGENT_JUPITER"/&gt;
      &lt;destinationAgent QMgr="QM_JUPITER" agent="AGENT_SATURN"/&gt;
      &lt;transferSet&gt;&lt;item checksumMethod="MD5" mode="binary"&gt;
        &lt;source disposition="leave" recursive="false"&gt;
          &lt;file&gt;/srv/nfs/incoming/*.txt&lt;/file&gt;&lt;/source&gt;
          &lt;destination exist="error" type="directory"&gt;
            &lt;file&gt;/srv/backup&lt;/file&gt;&lt;/destination&gt;
          &lt;/item&gt;&lt;/transferSet&gt;&lt;/managedTransfer&gt;&lt;/request&gt;
        &lt;/originalRequest>
      <updatedRequest>&lt;?xml version="1.0" encoding="UTF-8"?&gt;&lt;request
        xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance" version="4.00"
        xsi:noNamespaceSchemaLocation="FileTransfer.xsd"&gt;&lt;managedTransfer&gt;
          &lt;originator&gt;&lt;hostName&gt;example.com.&lt;/hostName&gt;
          &lt;userID&gt;mqm&lt;/userID&gt;&lt;/originator&gt;
          &lt;sourceAgent QMgr="QM_JUPITER" agent="AGENT_JUPITER"/&gt;
          &lt;destinationAgent QMgr="QM_JUPITER" agent="AGENT_SATURN"/&gt;
          &lt;transferSet&gt;&lt;item checksumMethod="MD5" mode="binary"&gt;
            &lt;source disposition="leave" recursive="false"&gt;
              &lt;file&gt;/srv/nfs/incoming/*.txt&lt;/file&gt;
              &lt;/source&gt;&lt;destination exist="error" type="directory"&gt;
                &lt;file&gt;/srv/backup&lt;/file&gt;&lt;/destination&gt;
              &lt;/item&gt;&lt;/transferSet&gt;&lt;/managedTransfer&gt;&lt;/request&gt;
            &lt;/updatedRequest>
          </taskXMLRequest>
</monitorLog>

```

Message de journal de condition de contrôle non satisfaite

```

<?xml version="1.0" encoding="UTF-8"?>
<monitorLog xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  monitorName="MONITORONE"
  referenceId="414d51205553322e42494e44494e47538b0f404d09430020"
  xsi:noNamespaceSchemaLocation="MonitorLog.xsd">
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
    <mqmdUserID>mqm</mqmdUserID>
  </originator>
</references>

```

```

    <createRequest>414d51205553322e42494e444494e47538b0f404d09430020</createRequest>
  </references>
  <action time="2011-01-26T12:58:46Z">triggerNotSatisfied</action>
  <monitorAgent agent="US2.BINDINGS.FILE" QMgr="US2.BINDINGS"/>
  <status resultCode="0"/>
</monitorLog>

```

Message de journal supprimé du moniteur

```

<?xml version="1.0" encoding="UTF-8"?>
<lst:monitorList xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xmlns:lst="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition"
  version="4.00"
  agent="AGENT_JUPITER"
  monitor="MONITORONE"
  xsi:schemaLocation="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition
MonitorList.xsd">
  <status state="deleted"/>
  <configuration>
    <description/>
    <resources>
      <directory recursionLevel="0" id="">/srv/nfs/incoming</directory>
    </resources>
    <triggerMatch>
      <conditions>
        <condition>
          <name/>
          <resource id=""/>
          <fileMatch>
            <pattern>*.completed</pattern>
          </fileMatch>
        </condition>
      </conditions>
    </triggerMatch>
    <tasks>
      <task>
        <name/>
        <description/>
        <taskXML>&lt;?xml version="1.0" encoding="UTF-8"?&gt;&lt;request
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance" version="4.00"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd"&gt;&lt;managedTransfer&gt;
    &lt;originator&gt;&lt;hostName&gt;example.ibm.com.&lt;/hostName&gt;
    &lt;userID&gt;mqm&lt;/userID&gt;&lt;/originator&gt;
    &lt;sourceAgent QMgr="QM_JUPITER" agent="AGENT_JUPITER"/&gt;
    &lt;destinationAgent QMgr="QM_JUPITER" agent="AGENT_SATURN"/&gt;
    &lt;transferSet&gt;&lt;item checksumMethod="MD5" mode="binary"&gt;
    &lt;source disposition="leave" recursive="false"&gt;
    &lt;file&gt;/srv/nfs/incoming/*.txt&lt;/file&gt;&lt;/source&gt;
    &lt;destination exist="error" type="directory"&gt;
    &lt;file&gt;/srv/backup&lt;/file&gt;&lt;/destination&gt;
    &lt;/item&gt;&lt;/transferSet&gt;&lt;/managedTransfer&gt;&lt;/request&gt;
        </taskXML>
      </task>
    </tasks>
  </configuration>
  <pollInterval units="minutes">1</pollInterval>
  <batch maxSize="1"/>
</lst:monitorList>

```

Format de message de demande de transfert de fichier

Les transferts de fichiers sont initiés par les messages XML arrivant dans une file d'attente de commandes d'agent, généralement à la suite de l'émission d'une commande de transfert de fichiers par un utilisateur ou à l'aide de IBM MQ Explorer. Le XML de la demande de transfert doit être conforme au schéma FileTransfer.xsd et avoir l'élément < request> comme élément racine. Le document de schéma FileTransfer.xsd se trouve dans le répertoire MQ_INSTALLATION_PATH/mqft/samples/schema. Le FileTransfer.xsd schéma importe fteutils.xsd, qui se trouve dans le même répertoire.

Les messages de transfert de fichier peuvent comporter l'un des trois éléments racine suivants:

- < request>-pour les nouvelles demandes de transfert de fichiers, les demandes d'appels gérés ou la suppression de transferts planifiés en attente
- < cancel>-pour l'annulation des transferts de fichiers en cours

- <transferSpecifications>-pour spécifier plusieurs groupes de fichiers de transfert, utilisés par la commande **fteCreateTransfer**

Pour plus d'informations sur la spécification de plusieurs groupes de transfert à l'aide de l'élément <transferSpecifications>, voir [Utilisation de fichiers de définition de transfert](#).

Schéma

Le schéma suivant décrit les éléments valides dans un message XML de demande de transfert.

```
<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema">
<xsd:include schemaLocation="fteutils.xsd"/>

<!--
  Defines the request of a managed transfer and version number
  <request version="1.00" ...
    <managedTransfer>
      ...
    </managedTransfer>
  </request>
-->
<xsd:element name="request">
  <xsd:complexType>
    <xsd:choice>
      <xsd:element name="managedTransfer" type="managedTransferType"/>
      <xsd:element name="deleteScheduledTransfer" type="deleteScheduledTransferType"/>
      <xsd:element name="managedCall" type="managedCallType"/>
    </xsd:choice>
    <xsd:attribute name="version" type="versionType" use="required"/>
  </xsd:complexType>
</xsd:element>

<!--
  Defines the cancel request of a managed transfer and version number
  <cancel version="1.00"
    xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
    <originator>
      <hostName>myMachine</hostName>
      <userID>myUserId</userID>
    </originator> - Delete a scheduled transfer.

    <transfer>
      Transfer ID to Cancel
    </transfer>
  </cancel>
-->
<xsd:element name="cancel">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="originator" type="hostUserIDType" maxOccurs="1" minOccurs="1"/>
      <xsd:choice>
        <xsd:element name="transfer" type="IDType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="call" type="IDType" maxOccurs="1" minOccurs="1"/>
      </xsd:choice>
      <xsd:element name="reply" type="replyType" maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="version" type="versionType" use="required"/>
  </xsd:complexType>
</xsd:element>

<!--
  Defines the transfer definition element structure.
  <transferSpecifications>
    <item ...
    <item ...
  </transferSpecifications>
-->
<xsd:element name="transferSpecifications">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="item" type="itemType" minOccurs="1" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<!--
```

Define a managed transfer of an instigator and request

```
<managedTransfer>
  <originator>
    ...
  </originator>
  <schedule>
    <submit timebase="source"|"UTC">2008-12-07T16:07</submit>
    <repeat>
      <frequency interval="hours">2</frequency>
      <expireTime>2008-12-0816:07</expireTime>
    </repeat>
  </schedule>
  <sourceAgent agent="here" QMgr="near"/>
  <destinationAgent agent="there" QMgr="far"/>
  <trigger>
    ...
  </trigger>
  <transferSet>
    ...
  </transferSet>
</managedTransfer>
-->
```

```
<xsd:complexType name="managedTransferType">
  <xsd:sequence>
    <xsd:element name="originator" type="origTransferRequestType" maxOccurs="1"
minOccurs="1"/>
    <xsd:element name="schedule" type="scheduleType" maxOccurs="1" minOccurs="0"/>
    <xsd:element name="sourceAgent" type="agentType" maxOccurs="1" minOccurs="1"/>
    <xsd:element name="destinationAgent" type="agentClientType" maxOccurs="1" minOccurs="1"/>
    <xsd:element name="trigger" type="triggerType" maxOccurs="1" minOccurs="0"/>
    <xsd:element name="reply" type="replyType" maxOccurs="1" minOccurs="0"/>
    <xsd:element name="transferSet" type="transferSetType" maxOccurs="1" minOccurs="1"/>
    <xsd:element name="job" type="jobType" maxOccurs="1" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<!--
  This is a modified form of origRequestType which is used on a managed transfer request.
  The hostName and userID are mandatory attributes in this case.
-->
```

```
<xsd:complexType name="origTransferRequestType">
  <xsd:sequence>
    <xsd:element name="hostName" type="xsd:string" minOccurs="1" maxOccurs="1"/>
    <xsd:element name="userID" type="xsd:string" minOccurs="1" maxOccurs="1"/>
    <xsd:element name="mqmdUserID" type="xsd:string" minOccurs="0" maxOccurs="1"/>
    <xsd:element name="webBrowser" type="xsd:string" minOccurs="0" maxOccurs="1"/>
    <xsd:element name="webUserID" type="xsd:string" minOccurs="0" maxOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<!--
  Defines the transferset of source and destination agent and one or more files
-->
```

```
<transferset priority="1">
  <metaDataSet>
    <metaData key="keyname">keyvalue</metaData>
    <metaData key="keyname">keyvalue</metaData>
  </metaDataSet>
```

```
  <item>
    ...
  </item>
</transferset>
```

```
-->
<xsd:complexType name="transferSetType">
  <xsd:sequence>
    <xsd:element name="metaDataSet" type="metaDataSetType" maxOccurs="1" minOccurs="0"/>
minOccurs="0"/>
    <xsd:element name="preSourceCall" type="commandActionType" maxOccurs="1"
minOccurs="0"/>
    <xsd:element name="postSourceCall" type="commandActionType" maxOccurs="1"
minOccurs="0"/>
    <xsd:element name="preDestinationCall" type="commandActionType" maxOccurs="1"
minOccurs="0"/>
    <xsd:element name="postDestinationCall" type="commandActionType" maxOccurs="1"
minOccurs="0"/>
    <xsd:element name="item" type="itemType" maxOccurs="unbounded" minOccurs="0"/>
  </xsd:sequence>
```

```

    <xsd:attribute name="priority" type="priorityType" use="optional"/>
</xsd:complexType>

<!--
  Define a file pair with source and destination
  <item mode=[binary|text]>
    <source recursive="false" disposition="leave">
      <file>filename</file>
    </source>

    <destination type="file" exist="error">
      <file>filename</file>
    </destination>

  </item>
-->
<xsd:complexType name="itemType">
  <xsd:sequence>
    <xsd:element name="source" type="fileSourceType" maxOccurs="1" minOccurs="1"/>
    <xsd:element name="destination" type="fileDestinationType" maxOccurs="1" minOccurs="1"/>
  </xsd:sequence>
  <xsd:attribute name="mode" type="modeType" use="required"/>
  <xsd:attribute name="checksumMethod" type="checkSumMethod" use="required"/>
</xsd:complexType>

<!--
  Defines the request to delete scheduled file transfer.
  <deleteScheduledTransfer>
    <originator>
      <delete>
        <hostName>myMachine</hostName>
        <userID>myUserId</userID>
      </delete>
    </originator>
    <ID>56</ID>
  </deleteScheduledTransfer>
-->
<xsd:complexType name="deleteScheduledTransferType">
  <xsd:sequence>
    <xsd:element name="originator" type="origDeleteType" maxOccurs="1" minOccurs="1"/>
    <xsd:element name="ID" type="idType" maxOccurs="1" minOccurs="1"/>
    <xsd:element name="reply" type="replyType" maxOccurs="1" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="managedCallType">
  <xsd:sequence>
    <xsd:element name="originator" type="origRequestType" maxOccurs="1" minOccurs="1"/>
    <xsd:element name="agent" type="agentType" maxOccurs="1" minOccurs="1"/>
    <xsd:element name="reply" type="replyType" maxOccurs="1" minOccurs="0"/>
    <xsd:element name="transferSet" type="callTransferSetType" maxOccurs="1" minOccurs="1"/>
    <xsd:element name="job" type="jobType" maxOccurs="1" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="callTransferSetType">
  <xsd:sequence>
    <xsd:element name="metaDataSet" type="metaDataSetType" maxOccurs="1" minOccurs="0"/>
    <xsd:element name="call" type="commandActionType" maxOccurs="1" minOccurs="1"/>
  </xsd:sequence>
  <xsd:attribute name="priority" type="priorityType" use="optional"/>
</xsd:complexType>
</xsd:schema>

```

Description du message de demande de transfert

Les éléments et attributs utilisés dans les messages de demande de transfert sont décrits dans la liste suivante:

Descriptions des éléments

<request>

Élément de groupe contenant tous les éléments requis pour spécifier une demande de transfert de fichier.

Attribut	Description
version	Indique la version de cet élément fournie par Managed File Transfer.

<managedTransfer>

Élément de groupe contenant tous les éléments requis pour un transfert de fichier unique ou un groupe de transferts de fichier unique.

< TransfertdeleteScheduled>

Élément de groupe contenant les informations d'origine et d'ID permettant d'annuler un transfert de planification.

<managedCall>

Élément de groupe qui contient tous les éléments requis pour un seul appel géré d'un programme ou d'un exécutable.

< ID>

Identificateur unique qui spécifie la demande de transfert à supprimer de la liste des transferts planifiés en attente.

< originaire>

Élément de groupe qui contient les éléments spécifiant l'émetteur de la demande.

<hostName>

Nom d'hôte du système sur lequel se trouve le fichier source.

<userID>

ID utilisateur à l'origine du transfert de fichiers.

< IDmqmdUser>

Facultatif. ID utilisateur IBM MQ fourni dans le descripteur de message (MQMD).

< planification>

Élément de groupe décrivant l'heure planifiée pour le transfert de fichier, le comportement de répétition et la date d'échéance de la prochaine occurrence.

< soumettre>

Indique la date et l'heure auxquelles le transfert planifié doit commencer.

Attribut	Description
base de temps	Indique le fuseau horaire à utiliser. Cet attribut peut adopter l'une des valeurs suivantes : <ul style="list-style-type: none"> • source-utiliser le fuseau horaire de l'agent source • admin-utilisez le fuseau horaire de l'administrateur qui a émis la commande • UTC-utiliser le temps universel coordonné
fuseau horaire	Description du fuseau horaire en fonction de la valeur de base de temps

< répétition>

Élément de groupe qui contient des détails sur la fréquence de répétition d'un transfert planifié, le nombre de répétitions d'un transfert planifié et le moment où un transfert planifié cesse de se répéter.

< fréquence>

Période qui doit s'écouler avant que le transfert ne se répète.

Attribut	Description
intervalle	Les unités d'intervalle, qui doivent être l'une des valeurs suivantes: <ul style="list-style-type: none"> • minutes • heures • jours • semaines • mois • années

<expireTime>

Élément facultatif indiquant la date et l'heure d'arrêt d'un transfert planifié répétitif. Cet élément et l'élément <expireCount> s'excluent mutuellement.

<expireCount>

Élément facultatif qui indique le nombre de fois où le transfert de fichiers planifié se produit avant l'arrêt. Cet élément et l'élément <expireTime> s'excluent mutuellement.

<sourceAgent>

Indique le nom de l'agent sur le système où se trouve le fichier source.

Attribut	Description
Agent	Indique le nom de l'agent.
QMgr	Nom du gestionnaire de files d'attente d'agent.

<destinationAgent>

Indique le nom de l'agent sur le système vers lequel vous souhaitez transférer le fichier.

Attribut	Description
Agent	Indique le nom de l'agent.
QMgr	Nom du gestionnaire de files d'attente d'agent.
hostName	Nom d'hôte ou adresse IP du gestionnaire de files d'attente de l'agent.
portNumber	Numéro de port utilisé pour les connexions client au gestionnaire de files d'attente de l'agent de destination.
canal	Nom de canal utilisé pour la connexion au gestionnaire de files d'attente de l'agent de destination.

< déclencheur >

Élément facultatif qui spécifie une condition qui doit être vraie pour que le transfert de fichier ait lieu.

Attribut	Description
journal	Indicateur signalant si les échecs de déclenchement sont consignés. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> • yes-des entrées de journal sont créées pour les transferts déclenchés ayant échoué • no-les entrées de journal ne sont pas créées pour les transferts déclenchés ayant échoué

<fileExist>

Indique une liste séparée par des virgules de noms de fichiers situés sur le même système que l'agent source. Si un fichier de cette liste de noms remplit la condition du déclencheur, le transfert est effectué. Cet élément et l'élément <fileSize> s'excluent mutuellement.

Attribut	Description
comparaison	Indique comment évaluer les noms de fichier source par rapport à la liste de noms. Les valeurs admises sont les suivantes : <ul style="list-style-type: none">• = au moins un nom de fichier dans la liste de noms doit correspondre• != un minimum de l'un des fichiers de la liste de noms n'existe pas
valeur	Indique le type de comparaison: <ul style="list-style-type: none">• exist: le fichier doit exister

<fileSize>

Indique une liste séparée par des virgules de noms de fichiers situés sur le même système que l'agent source. Si un fichier de cette liste de noms remplit la condition du déclencheur, le transfert est effectué. Cet élément et l'élément <fileExist> s'excluent mutuellement.

Attribut	Description
comparaison	Indique comment évaluer les noms de fichier source par rapport à la liste de noms. La valeur valide est la suivante: <ul style="list-style-type: none">• > = l'un des noms de fichier de la liste de noms existe et a une taille minimale spécifiée dans l'attribut de valeur
valeur	Taille de fichier indiquée sous la forme d'un entier avec les unités suivantes: <ul style="list-style-type: none">• B-octets• Ko-kiloctets• Mo-mégaoctets• Giga-Go (la valeur des unités n'est pas sensible à la casse)

< réponse>

Indique le nom de la file d'attente de réponses temporaire générée pour les transferts de fichiers synchrones (indiquée avec le paramètre **-w** sur la ligne de commande). Le nom de la file d'attente est défini par la clé **dynamicQueuePrefix** dans le fichier de configuration `command.properties` ou par la valeur par défaut `WMQFTE.*` Si aucune valeur n'est spécifiée,

Attribut	Description
détaillé	Indique si des informations détaillées sur les résultats de transfert sont requises dans le message de réponse. Plusieurs messages de réponse peuvent être générés pour chaque transfert. Les valeurs admises sont les suivantes : <ul style="list-style-type: none">• true-des informations de réponse détaillées sont requises. Le format des informations est le même que celui publié dans le journal de transfert dans les messages de progression, c'est-à-dire l'élément <transferSet>. Pour plus d'informations, voir la section «Formats des messages du journal de transfert de fichiers», à la page 771. Des informations de réponse détaillées sont disponibles uniquement lorsque la propriété <code>enableDetailedReplyMessages</code> est définie sur true pour l'agent source de transfert.• false-les informations de réponse détaillées ne sont pas requises. La valeur par défaut est false.

Attribut	Description
QMGR	Nom du gestionnaire de files d'attente de commandes sur lequel la file d'attente dynamique temporaire est générée pour recevoir les réponses.
persistant	Indique si le message écrit dans la file d'attente de réponses est persistant. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> • true-le message est persistant • false-le message n'est pas persistant • qdef-la persistance du message est définie par les propriétés de la file d'attente de réponses La valeur par défaut est false.

<transferSet>

Indique un groupe de transferts de fichiers que vous souhaitez effectuer ensemble ou un groupe d'appels gérés que vous souhaitez effectuer ensemble. Lors de la transmission, <transferSet> est un élément de groupe contenant des éléments < item>.

Attribut	Description
priorité	Niveau de priorité du transfert. La priorité est une valeur comprise entre 0 et 9, où 0 est la priorité la plus basse. Le niveau de priorité par défaut est 0 et, par défaut, le transfert utilise le niveau de priorité de l'agent source.

< Ensemble de métadonnéesmetaData>

Élément de groupe facultatif contenant un ou plusieurs éléments de métadonnées.

<metaData>

Indique les métadonnées définies par l'utilisateur qui sont transmises aux points d'exit appelés par l'agent. L'élément contient la valeur de métadonnées sous forme de chaîne.

Attribut	Description
clé	Nom de métadonnées sous forme de chaîne

< appel>

Élément de groupe qui contient des éléments < command> spécifiant le programme ou l'exécutable à appeler.

< commande>

Indique le programme ou l'exécutable à appeler. La commande doit se trouver dans le chemin de la commande de l'agent. Pour plus d'informations, voir [Tableau 45, à la page 566](#). Cet élément peut contenir des éléments facultatifs < argument>.

Attribut	Description
nom	Nom de la commande.
successRC	Code retour de réussite renvoyé par cette commande. La valeur par défaut est 0.
retryCount	Nombre de tentatives de relance de la commande en cas d'échec.
retryWait	Délai d'attente, en secondes, entre les nouvelles tentatives de la commande.
type	Type de programme à appeler. Les valeurs valides sont antscript, jcl ou exécutable.

<argument>

Indique un argument à transmettre à la commande.

<item>

Élément de groupe qui contient des éléments spécifiant les noms et les emplacements des fichiers source et de destination.

Attribut	Description
mode	Indique que le mode de transfert est binaire ou texte.
checksumMethod	Indique le type d'algorithme de hachage qui génère le résumé du message pour créer la signature numérique. Les valeurs valides sont MD5 ou aucune.

< source>

Élément de groupe qui spécifie les fichiers sur le système source et s'ils sont supprimés à la fin du transfert

Attribut	Description
recursive	Indique que les fichiers sont transférés de manière récursive dans des sous-répertoires lorsque l'élément < source> est un répertoire ou contient des caractères génériques.
disposition	Indique l'action effectuée sur l'élément < source> lorsque < source> a été transféré vers sa destination. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> • leave-les fichiers source restent inchangés. • delete-les fichiers source sont supprimés du système source après le transfert du fichier source.

<fichier>

Indique la source du transfert.  Pour Multiplatforms, la source de transfert peut être un fichier ou un nom de répertoire. Pour la plateforme z/OS, la source de transfert peut être un nom de fichier, de répertoire, de fichier ou de fichier partitionné. Utilisez le chemin qualifié complet dans le format correspondant à votre système d'exploitation, par exemple C: / from/here .txt. N'utilisez pas d'URI de fichier.

Attribut	Description
alias	Indique un alias pour le fichier source. Cet alias est le nom du fichier source, à l'exclusion de tout chemin de répertoire spécifié pour le transfert.
EOL	Indique le marqueur de fin de ligne pour les transferts de texte. Les valeurs admises sont : <ul style="list-style-type: none"> • Saut de ligne-caractère de saut de ligne uniquement • CRLF-retour chariot et séquence de caractères de saut de ligne
codage	Codage du fichier source pour un transfert de fichier texte.
 délimiteur	Indique le délimiteur qui est inclus entre les enregistrements dans les fichiers source orientés enregistrement, par exemple, les fichiers z/OS. Indiquez la valeur de délimiteur sous la forme de deux chiffres hexadécimaux dans la plage 00-FF, avec le préfixe x. Par exemple, x12 ou x03,x7F.
delimiterType	Indique le type de délimiteur inclus dans le fichier de destination après les données de message individuelles. Les valeurs valides sont les suivantes: <ul style="list-style-type: none"> • binary-délimiteur hexadécimal

Attribut	Description
	Cet attribut est disponible uniquement si vous avez activé la fonction 7.0.4.1 .
delimiterPosition	Indique la position à laquelle insérer des délimiteurs lors de l'écriture d'enregistrements de fichier source orientés enregistrement dans un fichier normal. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> • prefix-le délimiteur est inséré dans le fichier de destination avant les données de chaque enregistrement de fichier orienté enregistrement source. • postfix-le délimiteur est inséré dans le fichier de destination après les données de chaque enregistrement de fichier orienté enregistrement source.
includeDelimiterInFile	Indique s'il faut inclure un délimiteur entre les enregistrements dans les fichiers source orientés enregistrement.
 Espaces keepTrailing	Indique si les espaces de fin doivent être conservés dans les enregistrements source lus à partir d'un fichier de longueur fixe dans le cadre d'un transfert en mode texte. Par défaut, les espaces de fin sont supprimés. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> • true-les espaces de fin sont conservés dans les enregistrements source lus à partir d'un ensemble de données de format de longueur fixe • false-les espaces de fin sont supprimés des enregistrements source lus à partir d'un ensemble de données de format de longueur fixe

< file_attente >

Lorsqu'il est utilisé avec l'élément < source >, indique le nom de la file d'attente à partir de laquelle effectuer le transfert, qui doit se trouver sur le gestionnaire de files d'attente de l'agent source. Utilisez le format *QUEUE*. N'incluez pas le nom du gestionnaire de files d'attente, la file d'attente doit être présente sur le gestionnaire de files d'attente de l'agent source. Vous ne pouvez pas utiliser l'élément < queue > dans l'élément < source >, si vous l'avez utilisé dans l'élément < destination >.

Attribut	Description
useGroups	Indique s'il faut transférer uniquement le premier groupe complet de messages de la file d'attente source. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> • true-transfère uniquement le premier groupe complet de messages • false-transfère tous les messages dans la file d'attente source
groupId	Indique le groupe de messages à lire dans la file d'attente source. Cet attribut est valide uniquement lorsque la valeur de l'attribut useGroups est true.
delimiterType	Indique le type de délimiteur inclus dans le fichier de destination après les données de message individuelles. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> • text-délimiteur de texte ou de littéral Java • binary-délimiteur hexadécimal
délimiteur	Indique le délimiteur inclus dans le fichier de destination entre les données de message individuelles.
delimiterPosition	Indique si le délimiteur est inclus dans le fichier de destination avant ou après les données de message individuelles. Les valeurs admises sont les suivantes :

Attribut	Description
	<ul style="list-style-type: none"> • prefix-le délimiteur est inclus avant les données • postfix-le délimiteur est inclus après les données
codage	Indique le codage de la file d'attente source.
waitTime	<p>Indique le délai d'attente, en secondes, de l'agent source pour l'un des éléments suivants:</p> <ul style="list-style-type: none"> • un message à afficher dans la file d'attente source, si la file d'attente est vide ou est devenue vide • un groupe complet à afficher dans la file d'attente source, si l'attribut useGroups a été défini sur true <p>Pour plus d'informations sur la définition de la valeur waitTime , voir «Conseils pour la spécification d'un temps d'attente sur un transfert de message vers fichier», à la page 704.</p>

< destination >

Élément de groupe qui spécifie la destination et le comportement si des fichiers existent au niveau de l'agent de destination.

Vous ne pouvez spécifier qu'un seul élément < file > et < queue > comme élément enfant de la destination.

Attribut	Description
type	<p>Type de destination. Les valeurs admises sont les suivantes :</p> <ul style="list-style-type: none"> • file -indique un fichier comme destination • directory -indique un répertoire comme destination •  dataset -spécifie un fichier z/OS comme destination •  pds -indique un fichier partitionné z/OS comme destination • queue -indique une file d'attente IBM MQ comme destination • filespace -indique un espace fichier comme destination <p>La valeur queue est valide uniquement lorsque l'élément < destination > a un élément enfant < queue >.</p> <p>La valeur filespace est valide uniquement lorsque l'élément < destination > a un élément enfant de < filespace >.</p> <p>Les autres valeurs sont valides uniquement lorsque l'élément < destination > a un élément enfant < fichier >.</p>
existe	<p>Indique l'action effectuée si un fichier de destination existe sur le système de destination. Les valeurs admises sont les suivantes :</p> <ul style="list-style-type: none"> • error-signe une erreur et le fichier n'est pas transféré. • overwrite-écrase le fichier de destination existant. <p>Cet attribut n'est pas valide si l'élément < destination > a un élément enfant < queue > ou < filespace >.</p>

<fichier>

Spécifie des paramètres supplémentaires pour l'élément **< destination >** décrit précédemment. Utilisez le chemin qualifié complet dans le format correspondant à votre système d'exploitation, par exemple C : /from/here .txt. N'utilisez pas d'URI de fichier.

Attribut	Description
alias	Indique un alias pour le fichier < destination > . Cet alias est le nom du fichier source, à l'exclusion de tout chemin de répertoire spécifié pour le transfert.
codage	Codage du fichier < destination > pour un transfert de fichier texte.
EOL	Indique le marqueur de fin de ligne pour les transferts de texte. Les valeurs admises sont : <ul style="list-style-type: none">• Saut de ligne-caractère de saut de ligne uniquement• CRLF-retour chariot et séquence de caractères de saut de ligne
truncateRecords	Facultatif. Indique que les enregistrements < destination > plus longs que l'attribut de fichier LRECL sont tronqués. <ul style="list-style-type: none">• True-les enregistrements sont tronqués• False-les enregistrements sont encapsulés Par défaut, le paramètre a la valeur false.

< file_attente >

Lorsqu'il est utilisé avec l'élément **< destination >** , indique le nom de la file d'attente vers laquelle effectuer le transfert, qui peut se trouver sur tout gestionnaire de files d'attente connecté au gestionnaire de files d'attente de l'agent de destination. Utilisez le format *QUEUE@QM* où *QUEUE* est le nom de la file d'attente dans laquelle placer les messages et *QM* est le gestionnaire de files d'attente dans lequel se trouve la file d'attente. Vous ne pouvez pas utiliser l'élément **< queue >** dans l'élément **< destination >**, si vous l'avez utilisé dans l'élément **< source >**.

Attribut	Description
délimiteur	Délimiteur permettant de fractionner le fichier en plusieurs messages.
delimiterType	Indique le type de délimiteur. Les valeurs admises sont les suivantes : <ul style="list-style-type: none">• text-expression régulière Java• binary-une séquence d'octets hexadécimaux• size-nombre d'octets, de kioctets ou de mébioctets. Par exemple, 1 B, 1 K ou 1 M.
delimiterPosition	Indique si le délimiteur est attendu avant ou après les données à inclure dans les messages individuels. Les options valides sont les suivantes: <ul style="list-style-type: none">• prefix-le délimiteur est attendu avant les données• postfix-le délimiteur est attendu après les données
includeDelimiterInMessage	Valeur booléenne indiquant s'il faut inclure les délimiteurs utilisés pour fractionner le fichier en plusieurs messages à la fin des messages.
codage	Indique le codage de la file d'attente de destination.
persistant	Indique si les messages sont persistants. Les valeurs admises sont les suivantes : <ul style="list-style-type: none">• true-les messages sont persistants

Attribut	Description
	<ul style="list-style-type: none"> • false-les messages ne sont pas persistants • qdef-la valeur de persistance des messages est définie par les paramètres de la file d'attente de destination
setMqProps	Valeur booléenne indiquant si les propriétés de message IBM MQ sont définies sur le premier message d'un fichier et sur tous les messages écrits dans la file d'attente lorsqu'une erreur se produit.
Page unrecognizedCode	<p>Indique si un transfert en mode texte échoue ou si la conversion est effectuée si la page de codes des données n'est pas reconnue par le gestionnaire de files d'attente de destination. Les valeurs admises sont les suivantes :</p> <ul style="list-style-type: none"> • fail-le transfert signale un échec • binary-les données sont converties dans la page de codes de destination et l'en-tête de message IBM MQ décrivant le format des données est défini sur MQFMT_NONE. <p>Le comportement par défaut est fail.</p>

< espace_fichier >

Élément de groupe spécifiant le nom de l'espace fichier vers lequel effectuer le transfert.

< nom >

Lorsqu'il est utilisé avec l'élément < filespace >, la valeur de cet élément indique le nom de l'espace fichier.

< appellepreSource >

Élément de groupe spécifiant une commande à appeler à la source du transfert, avant le début du transfert.

< appellepostSource >

Élément de groupe spécifiant une commande à appeler à la source du transfert, une fois le transfert terminé.

< appellepreDestination >

Élément de groupe spécifiant une commande à appeler à la destination du transfert, avant le début du transfert.

< appellepostDestination >

Élément de groupe spécifiant une commande à appeler à la destination du transfert, une fois le transfert terminé.

< commande >

Lorsqu'il est utilisé avec l'élément < preSourceCall >, < postSourceCall >, < preDestinationCall > ou < postDestinationCall >, cet élément spécifie la commande à appeler. La commande doit se trouver dans le chemin de la commande de l'agent. Pour plus d'informations, voir [Tableau 45](#), à la page 566.

Attribut	Description
nom	Nom de la commande à exécuter.
successRC	Code retour attendu si la commande s'exécute correctement.

< argument >

Lorsqu'il est utilisé avec l'élément < command >, cet élément spécifie un argument à transmettre à la commande. Vous pouvez avoir n'importe quel nombre d'éléments < argument > dans un élément < command >.

< travail >

Élément de groupe facultatif contenant des informations de travail pour l'ensemble de la spécification de transfert. < job > est un identificateur de nom de travail défini par l'utilisateur qui est ajouté au message du journal lorsque le transfert a commencé. Cet élément < job > est identique à l'élément < job > qui apparaît dans le message du journal de transfert, décrit dans la rubrique suivante: [«Formats des messages du journal de transfert de fichiers»](#), à la page 771.

< nom >

Lorsqu'il est utilisé avec l'élément < job >, la valeur de cet élément indique le nom du travail.

<transferSpecifications >

Élément de groupe qui contient des éléments < item > pour plusieurs groupes de transfert. Pour plus d'informations sur l'utilisation de cet élément, voir [Utilisation des fichiers de définition de transfert](#) .

< annuler >

Élément de groupe contenant tous les éléments requis pour annuler un transfert de fichier en cours.

Attribut	Description
version	Indique la version de cet élément fournie par Managed File Transfer.

< transfer >

Lorsqu'il est utilisé avec l'élément < cancel >, la valeur de cet élément indique l'ID de demande de transfert à annuler.

< travail >

Élément de groupe contenant des informations sur le travail.

<jobName >

Indique l'identificateur de travail logique.

Format du message d'annulation du transfert de fichiers

Une demande de transfert de fichier renvoie un ID de 48 caractères qui identifie le transfert pour un agent spécifique. Cet ID est utilisé pour annuler les transferts.

Description du message d'annulation de transfert

Les éléments et attributs utilisés dans les messages d'annulation de transfert sont décrits ci-dessous:

< annuler >

Élément de groupe contenant tous les éléments requis pour annuler un transfert de fichier en cours.

Attribut	Description
version	Indique la version de cet élément fournie par Managed File Transfer.

< originaire >

Élément de groupe qui contient les éléments spécifiant l'émetteur de la demande.

<hostName >

Nom d'hôte du système sur lequel se trouve le fichier source.

<userID >

ID utilisateur à l'origine du transfert de fichiers.

< IDmqmdUser >

Facultatif. ID utilisateur IBM MQ fourni dans le descripteur de message (MQMD).

< transfer >

Lorsqu'il est utilisé avec l'élément < cancel >, la valeur de cet élément indique l'ID de demande de transfert à annuler.

< travail >

Facultatif. Élément de groupe contenant des informations sur le travail.

< jobName >

Indique l'identificateur de travail logique.

Exemples

Des exemples de messages XML conformes à ce schéma sont fournis pour chacune des demandes suivantes:

- [Créer un transfert de fichier](#)
- [Création d'une demande de transfert de fichier asynchrone](#)
- [Annuler un transfert de fichier](#)
- [Création d'un transfert planifié](#)
- [Suppression d'un transfert planifié](#)
- [Création d'un appel géré](#)
- [Création d'un transfert de fichiers incluant des appels gérés](#)

Exemples de message de demande de transfert de fichier

Exemples de messages que vous pouvez placer dans la file d'attente de commandes de l'agent pour demander à l'agent de créer ou d'annuler un transfert.

Créer une demande de transfert

```
<?xml version="1.0" encoding="UTF-8"?>
<request xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
version="4.00"
xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>example.com.</hostName>
      <userID>mqm</userID>
    </originator>
    <sourceAgent QMgr="QM_JUPITER" agent="AGENT_JUPITER"/>
    <destinationAgent QMgr="QM_JUPITER" agent="AGENT_JUPITER"/>
    <transferSet>
      <item checksumMethod="MD5" mode="binary">
        <source disposition="leave" recursive="false">
          <file>/etc/passwd</file>
        </source>
        <destination exist="overwrite" type="directory">
          <file>/tmp</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Créer une demande de transfert-synchrone

Lorsqu'un utilisateur demande une demande synchrone de blocage, c'est-à-dire qu'il attend la fin du transfert et qu'il reçoit des messages d'état, le message placé dans la file d'attente de commandes contient un élément de réponse qui spécifie la file d'attente à laquelle un message de réponse est envoyé. L'exemple suivant illustre le message placé dans la file d'attente de commandes utilisée par FTEAGENT:

```
<?xml version="1.0" encoding="UTF-8"?>
<request version="4.00"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
```

```

    xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
<managedTransfer>
  <originator>
    <hostName>reportserver.com</hostName>
    <userID>USER1</userID>
  </originator>
  <sourceAgent agent="FTEAGENT"
    QMgr="QM1"/>
  <destinationAgent agent="AGENT2"
    QMgr="QM2"/>
  <reply QMGR="QM1">WMQFTE.492D0D5502770020</reply>
  <transferSet>
    <item mode="binary" checksumMethod="MD5">
      <source recursive="false" disposition="leave">
        <file>c:\sourcefiles\source1.doc</file>
      </source>
      <destination type="file" exist="overwrite">
        <file>c:\destinationfiles\dest1.doc</file>
      </destination>
    </item>
  </transferSet>
</managedTransfer>
</request>

```

L'élément < reply> est rempli avec le nom du gestionnaire de files d'attente de commandes dans lequel une file d'attente dynamique temporaire a été créée pour recevoir une réponse indiquant que le transfert a abouti (ou non). Le nom de la file d'attente dynamique temporaire se compose de deux parties:

- Préfixe tel que défini par la clé **dynamicQueuePrefix** dans le fichier de configuration `command.properties` (il s'agit de `WMQFTE`), par défaut)
- ID de la file d'attente généré par IBM MQ

Demande d'annulation de transfert

```

<?xml version="1.0" encoding="UTF-8"?>
<cancel xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
  </originator>
  <transfer>414D51205553322E42494E44494E47538B0F404D032C0020</transfer>
  <reply QMGR="QM_JUPITER">WMQFTE.4D400F8B20002007</reply>
</cancel>

```

Référence associée

«Format de message de demande de transfert de fichier», à la page 804

Les transferts de fichiers sont initiés par les messages XML arrivant dans une file d'attente de commandes d'agent, généralement à la suite de l'émission d'une commande de transfert de fichiers par un utilisateur ou à l'aide de IBM MQ Explorer. Le XML de la demande de transfert doit être conforme au schéma `FileTransfer.xsd` et avoir l'élément < request> comme élément racine. Le document de schéma `FileTransfer.xsd` se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/schema`. Le `FileTransfer.xsd` schéma importe `fteutils.xsd`, qui se trouve dans le même répertoire.

Exemples de message de transfert de fichier planifié

Exemples de messages que vous pouvez placer dans la file d'attente de commandes de l'agent pour demander à l'agent de créer ou de supprimer une planification.

Créer un transfert planifié

```

<?xml version="1.0" encoding="UTF-8"?>
<request xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>example.com.</hostName>
      <userID>mqm</userID>
    </originator>
  </managedTransfer>
</request>

```

```

</originator>
<schedule>
  <submit timebase="admin" timezone="Europe/London">2010-01-01T21:00</submit>
</schedule>
<sourceAgent QMgr="US2.BINDINGS" agent="US2.BINDINGS.FILE"/>
<destinationAgent QMgr="US2.BINDINGS" agent="US2.BINDINGS.FILE"/>
<transferSet>
  <item checksumMethod="MD5" mode="binary">
    <source disposition="leave" recursive="false">
      <file>/etc/passwd</file>
    </source>
    <destination exist="overwrite" type="directory">
      <file>/tmp</file>
    </destination>
  </item>
</transferSet>
</managedTransfer>
</request>

```

Supprimer le transfert planifié

```

<?xml version="1.0" encoding="UTF-8"?>
<request xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <deleteScheduledTransfer>
    <originator>
      <delete>
        <hostName>example.com.</hostName>
        <userID>mqm</userID>
      </delete>
    </originator>
    <ID>1</ID>
    <reply QMGR="US2.BINDINGS">WMQFTE.4D400F8B20003902</reply>
  </deleteScheduledTransfer>
</request>

```

Référence associée

«Format de message de demande de transfert de fichier», à la page 804

Les transferts de fichiers sont initiés par les messages XML arrivant dans une file d'attente de commandes d'agent, généralement à la suite de l'émission d'une commande de transfert de fichiers par un utilisateur ou à l'aide de IBM MQ Explorer. Le XML de la demande de transfert doit être conforme au schéma FileTransfer.xsd et avoir l'élément <request> comme élément racine. Le document de schéma FileTransfer.xsd se trouve dans le répertoire MQ_INSTALLATION_PATH/mqft/samples/schema. Le FileTransfer.xsd schéma importe fteutils.xsd, qui se trouve dans le même répertoire.

Exemples de message de demande d'appel d'agent MFT

Exemples de messages que vous pouvez placer dans la file d'attente de commandes de l'agent pour demander à l'agent de créer un appel géré ou un transfert qui appelle des programmes.

Exemple de demande d'appel géré

```

<?xml version="1.0" encoding="UTF-8"?>
<request xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="1.00"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedCall>
    <originator>
      <hostName>example.com.</hostName>
      <userID>mqm</userID>
    </originator>
    <agent agent="DNWE" QMgr="QM1"/>
    <transferSet>
      <call>
        <command name="echo" successRC="0">
          <argument>call</argument>
          <argument>test</argument>
        </command>
      </call>
    </transferSet>
  </managedCall>
</request>

```

```

    <job>
      <name>managedCallCalls.xml</name>
    </job>
  </managedCall>
</request>

```

Exemple de demande de transfert géré avec des appels

```

<?xml version="1.0" encoding="UTF-8"?>
<request xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="1.00"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>example.com.</hostName>
      <userID>mqm</userID>
    </originator>
    <sourceAgent agent="DNWE" QMgr="QM1"/>
    <destinationAgent agent="DNWE" QMgr="QM1"/>
    <transferSet>
      <preSourceCall>
        <command name="echo" successRC="0">
          <argument>preSourceCall</argument>
          <argument>test</argument>
        </command>
      </preSourceCall>
      <postSourceCall>
        <command name="echo" successRC="0">
          <argument>postSourceCall</argument>
          <argument>test</argument>
        </command>
      </postSourceCall>
      <preDestinationCall>
        <command name="echo" successRC="0">
          <argument>preDestinationCall</argument>
          <argument>test</argument>
        </command>
      </preDestinationCall>
      <postDestinationCall>
        <command name="echo" successRC="0">
          <argument>postDestinationCall</argument>
          <argument>test</argument>
        </command>
      </postDestinationCall>
    </transferSet>
  </job>
  <name>managedTransferCalls.xml</name>
</job>
</managedTransfer>
</request>

```

Concepts associés

[«Spécification des programmes à exécuter avec MFT», à la page 256](#)

Vous pouvez exécuter des programmes sur un système sur lequel un Managed File Transfer Agent est en cours d'exécution. Dans le cadre d'une demande de transfert de fichiers, vous pouvez spécifier un programme à exécuter avant ou après le début d'un transfert. En outre, vous pouvez démarrer un programme qui ne fait pas partie d'une demande de transfert de fichier en soumettant une demande d'appel géré.

Référence associée

[«Format de message de demande de transfert de fichier», à la page 804](#)

Les transferts de fichiers sont initiés par les messages XML arrivant dans une file d'attente de commandes d'agent, généralement à la suite de l'émission d'une commande de transfert de fichiers par un utilisateur ou à l'aide de IBM MQ Explorer. Le XML de la demande de transfert doit être conforme au schéma FileTransfer.xsd et avoir l'élément < request> comme élément racine. Le document de schéma FileTransfer.xsd se trouve dans le répertoire *MQ_INSTALLATION_PATH/mqft/samples/schema*. Le FileTransfer.xsd schéma importe fteutils.xsd, qui se trouve dans le même répertoire.

Formats des messages de demande du moniteur MFT

Les moniteurs de ressources sont créés lorsqu'un message XML approprié arrive dans la file d'attente de commandes d'un agent, généralement suite à l'émission par un utilisateur de la commande de surveillance fteCreateou à l'utilisation de l'interface IBM MQ Explorer .

Le code XML du moniteur doit être conforme au schéma Monitor.xsd en utilisant l'élément <monitor> comme élément racine.

Les messages du moniteur peuvent comporter l'un des éléments racine suivants:

- <monitor> -pour la création et le démarrage d'un nouveau moniteur de ressources
- <deleteMonitor> -pour l'arrêt et la suppression d'un moniteur existant

Il n'y a pas de message de commande pour la commande fteListMonitors car la commande extrait directement les définitions de moniteur correspondantes de SYSTEM.FTE .

Schéma

Le schéma suivant décrit les éléments valides dans un message XML de demande de moniteur.

```
<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema"
            targetNamespace="https://www.ibm.com/xmlns/wmqfte/7.0.1/
MonitorDefinition"
            xmlns="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition">

<xsd:include schemaLocation="FileTransfer.xsd"/>

  <xsd:element name="monitor">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="name" type="monitorNameType"
                    minOccurs="1" maxOccurs="1"/>
        <xsd:element name="description" type="xsd:string"
                    minOccurs="0" maxOccurs="1"/>
        <xsd:element name="pollInterval" type="pollIntervalType"
                    minOccurs="1" maxOccurs="1"
default="10"/>
        <xsd:element name="batch" type="batchType"
                    minOccurs="0" maxOccurs="1"/>
        <xsd:element name="agent" type="agentNameType"
                    minOccurs="1" maxOccurs="1"/>
        <xsd:element name="resources" type="monitorResourcesType"
                    minOccurs="0"
maxOccurs="1"/>
        <xsd:element name="triggerMatch" type="triggerMatchType"
                    maxOccurs="1" minOccurs="1"/>
        <xsd:element name="reply" type="replyType"
                    maxOccurs="1" minOccurs="0"/>
        <xsd:element name="tasks" type="monitorTasksType"
                    maxOccurs="1" minOccurs="1"/>
        <xsd:element name="originator" type="origRequestType"
                    maxOccurs="1" minOccurs="1"/>
        <xsd:element name="job" type="jobType"
                    maxOccurs="1" minOccurs="0"/>
        <xsd:element name="defaultVariables" type="defaultVariablesType"
                    maxOccurs="1" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="version" type="versionType" use="required"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="deleteMonitor">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="name" type="monitorNameType"
                    minOccurs="1" maxOccurs="1"/>
        <xsd:element name="originator" type="origRequestType"
                    maxOccurs="1" minOccurs="1"/>
        <xsd:element name="reply" type="replyType"
                    maxOccurs="1" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="version" type="versionType" use="required"/>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

```

<xsd:complexType name="transferRequestType">
  <xsd:choice>
    <xsd:element name="managedTransfer" type="managedTransferType"/>
    <xsd:element name="managedCall" type="managedCallType"/>
  </xsd:choice>
  <xsd:attribute name="version" type="versionType"/>
</xsd:complexType>

<xsd:complexType name="monitorResourcesType">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="directory" type="monitoredDirectoryType"
        minOccurs="1" maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:element name="queue" type="monitoredQueueType"/>
  </xsd:choice>
</xsd:complexType>

<xsd:complexType name="monitoredDirectoryType">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute name="recursionLevel"
type="xsd:nonNegativeInteger"/>
      <xsd:attribute name="id" type="resourceIdAttrType"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="monitoredQueueType">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute name="id" type="resourceIdAttrType"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="triggerMatchType">
  <xsd:sequence>
    <xsd:element name="conditions" type="conditionsType"
      minOccurs="1" maxOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="conditionsType">
  <xsd:choice minOccurs="1">
    <xsd:element name="allOf" type="listPredicateType"
      minOccurs="1" maxOccurs="1"/>
    <xsd:element name="anyOf" type="listPredicateType"
      minOccurs="1" maxOccurs="1"/>
    <xsd:element name="condition" type="conditionType"
      minOccurs="1" maxOccurs="1"/>
  </xsd:choice>
</xsd:complexType>

<xsd:complexType name="listPredicateType">
  <xsd:choice>
    <xsd:element name="condition" type="conditionType"
      minOccurs="1" maxOccurs="unbounded"/>
  </xsd:choice>
</xsd:complexType>

<xsd:complexType name="conditionType">
  <xsd:sequence>
    <xsd:element name="name" type="conditionNameType"
      minOccurs="0" maxOccurs="1"/>
    <xsd:element name="resource" type="resourceIdType"
      minOccurs="0" maxOccurs="1"/>
    <xsd:choice minOccurs="1">
      <xsd:element name="fileMatch"
type="fileMatchConditionType"
      minOccurs="1" maxOccurs="1"/>
      <xsd:element name="fileNoMatch"
type="fileNoMatchConditionType"
      minOccurs="1"
maxOccurs="1"/>
      <xsd:element name="fileSize"
type="fileSizeConditionType"
      minOccurs="1" maxOccurs="1"/>
      <xsd:element name="queueNotEmpty"
type="queueNotEmptyConditionType"
      minOccurs="1" maxOccurs="1"/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>

```

```

        <xsd:element name="completeGroups"
type="completeGroupsConditionType"
        minOccurs="1"          maxOccurs="1"/>
        <xsd:element name="fileSizeSame"      type="fileSizeSameType"
        minOccurs="1"          maxOccurs="1"/>
    </xsd:choice>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="fileMatchConditionType">
    <xsd:sequence>
        <xsd:element name="pattern" type="conditionPatternType"
        minOccurs="0" default="*.*"/>
        <xsd:element name="exclude" type="conditionPatternType"
        minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="fileNoMatchConditionType">
    <xsd:sequence>
        <xsd:element name="pattern" type="conditionPatternType"
        minOccurs="0" default="*.*"/>
        <xsd:element name="exclude" type="conditionPatternType"
        minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="fileSizeConditionType">
    <xsd:sequence>
        <xsd:element name="compare" type="sizeCompareType"
        minOccurs="1" default="0"/>
        <xsd:element name="pattern" type="conditionPatternType"
        minOccurs="0" default="*.*"/>
        <xsd:element name="exclude" type="conditionPatternType"
        minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="sizeCompareType">
    <xsd:simpleContent>
        <xsd:extension base="xsd:int">
            <xsd:attribute name="operator" type="sizeOperatorType"
use="required"/>
            <xsd:attribute name="units" type="fileSizeUnitsType"
use="required"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:simpleType name="sizeOperatorType">
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value=">="/>
    </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="fileSizeUnitsType">
    <xsd:restriction base="xsd:string">
        <xsd:pattern value="[bB] | [kK] [bB] | [mM] [bB] | [gG] [bB]"/>
    </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="conditionPatternType">
    <xsd:simpleContent>
        <xsd:extension base="xsd:string">
            <xsd:attribute name="type" type="patternTypeAttributeType"
use="optional" default="wildcard"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:simpleType name="patternTypeAttributeType">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="regex"/>
        <xsd:enumeration value="wildcard"/>
    </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="conditionNameType">
    <xsd:simpleContent>
        <xsd:extension base="xsd:string"/>
    </xsd:simpleContent>
</xsd:complexType>

```

```

<xsd:complexType name="queueNotEmptyConditionType"/>
<xsd:complexType name="completeGroupsConditionType"/>
<xsd:complexType name="fileSizeSameType">
  <xsd:sequence>
    <xsd:element name="pattern" type="conditionPatternType"
      minOccurs="1" maxOccurs="1"/>
    <xsd:element name="exclude" type="conditionPatternType"
      minOccurs="0" maxOccurs="1"/>
  </xsd:sequence>
  <xsd:attribute name="polls" type="positiveIntegerType" use="required"/>
</xsd:complexType>
<xsd:complexType name="pollIntervalType">
  <xsd:simpleContent>
    <xsd:extension base="xsd:int">
      <xsd:attribute name="units" type="timeUnitsType"
        use="optional" default="minutes"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
<xsd:complexType name="batchType">
  <xsd:attribute name="maxSize" type="positiveIntegerType" use="required"/>
</xsd:complexType>
<xsd:simpleType name="timeUnitsType">
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="seconds"/>
    <xsd:enumeration value="minutes"/>
    <xsd:enumeration value="hours"/>
    <xsd:enumeration value="days"/>
    <xsd:enumeration value="weeks"/>
    <xsd:enumeration value="months"/>
    <xsd:enumeration value="years"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="monitorTasksType">
  <xsd:sequence>
    <xsd:element name="task" type="monitorTaskType"
      minOccurs="1" maxOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="monitorTaskType">
  <xsd:sequence>
    <xsd:element name="name" type="monitorTaskNameType"
      minOccurs="1" maxOccurs="1"/>
    <xsd:element name="description" type="xsd:string"
      minOccurs="0" maxOccurs="1"/>
    <xsd:element name="transfer" type="transferTaskType"
      minOccurs="0" maxOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="transferTaskType">
  <xsd:sequence>
    <xsd:element name="request" type="transferRequestType"
      minOccurs="1" maxOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="resourceIdType">
  <xsd:attribute name="id" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:simpleType name="resourceIdAttrType">
  <xsd:restriction base="xsd:string"></xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="monitorNameType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="^[%]*"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="agentNameType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[%_0-9A-Z]*"/>
  </xsd:restriction>
</xsd:simpleType>

```

```

    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="monitorTaskNameType">
    <xsd:restriction base="xsd:string">
      <xsd:pattern value=".*"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name="defaultVariablesType">
    <xsd:sequence>
      <xsd:element name="variable" type="variableType"
        maxOccurs="unbounded" minOccurs="1"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="variableType">
    <xsd:simpleContent>
      <xsd:extension base="xsd:string">
        <xsd:attribute name="key" type="xsd:string" use="required"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
</xsd:schema>

```

Description du message de création de moniteur

Les éléments et attributs utilisés dans les messages de création de moniteur sont décrits ci-dessous:

Descriptions des éléments

<monitor>

Élément de groupe contenant tous les éléments requis pour annuler un transfert de fichier en cours.

Attribut	Description
version	Indique la version de cet élément fournie par Managed File Transfer.

<nom>

Nom du moniteur, unique dans l'agent du moniteur.

<description>

Description du moniteur (non utilisé actuellement).

<pollInterval>

Intervalle de temps entre chaque vérification de la ressource par rapport à la condition de déclenchement.

Attribut	Description
unités	Indique les unités de temps de l'intervalle d'interrogation. Les valeurs admises sont : <ul style="list-style-type: none"> • secondes • minutes • heures • jours • semaines • mois • années

<agent>

Nom de l'agent auquel le moniteur est associé.

<Resources>

Élément de groupe qui contient les éléments spécifiant les ressources à surveiller.

<répertoire>

Chemin qualifié complet spécifiant le répertoire sur la machine de l'agent du moniteur à surveiller.

Attribut	Description
recursionLevel	Nombre de sous-répertoires à surveiller en plus du répertoire spécifié.
ID	Identificateur unique de la ressource.

< file_attente>

Nom de la file d'attente spécifiant la file d'attente à surveiller sur le gestionnaire de files d'attente de l'agent de surveillance.

<triggerMatch>

Élément de groupe qui contient les éléments spécifiant les conditions de déclenchement à comparer à la ressource contrôlée.

<conditions>

Élément de groupe qui contient les éléments spécifiant le type de condition à comparer à la ressource contrôlée.

<allOf>

Prédicat qui indique que toutes les conditions contenues doivent être satisfaites.

<anyOf>

Prédicat qui indique que toutes les conditions contenues doivent être satisfaites.

< condition>

Définit une condition de comparaison qui contribuera à la condition de déclenchement globale du moniteur.

< nom>

Nom de la condition.

<resource>

Identifie la définition de ressource à laquelle la condition doit être comparée.

Attribut	Description
ID	Identificateur unique de la ressource.

Si la ressource surveillée est un répertoire, l'un des trois éléments suivants doit être spécifié dans la condition:

- fileMatch
- fileNocorrespondance
- fileSize

Si la ressource surveillée est une file d'attente, l'un des deux éléments suivants doit être spécifié dans la condition:

- queueNotvide
- completeGroups

<fileMatch>

Élément de groupe pour une condition de correspondance de nom de fichier.

< modèle>

Indique un modèle de correspondance de nom de fichier. Les fichiers de la ressource doivent correspondre au modèle afin de satisfaire à la condition. Le modèle par défaut est * (tout fichier correspond).

<fileNoCorrespondance >

Élément de groupe pour une condition de correspondance de nom de fichier inverse.

< modèle >

Indique un modèle de correspondance de nom de fichier inverse. Si aucun fichier de la ressource contrôlée ne correspond, la condition est satisfaite. Le modèle par défaut est * (l'absence de fichier correspond).

< fileSize >

Élément de groupe pour une comparaison de taille de fichier.

< compare >

Indique une comparaison de taille de fichier. Cette valeur doit être un nombre entier non négatif.

Attribut	Description
opérateur	Opérateur de comparaison à utiliser. Seul > = est pris en charge.
unités	Indique les unités de taille de fichier, qui peuvent être l'une des suivantes: <ul style="list-style-type: none"> • B-octets • Ko-kilooctets • Mo-mégaoctets • Giga-Go La valeur des unités est insensible à la casse, donc mb'fonctionne aussi bien que MB'.

< modèle >

Modèle de nom de fichier à mettre en correspondance. La valeur par défaut est * (tout fichier correspond).

< queueNotvide >

Cela ne peut être spécifié que si la ressource est une file d'attente. Indique qu'un message doit figurer dans la file d'attente pour que le moniteur soit déclenché.

< completeGroups >

Cela ne peut être spécifié que si la ressource est une file d'attente. Indique qu'un groupe complet de messages doit être présent dans la file d'attente pour que le moniteur soit déclenché. Une tâche de transfert unique est exécutée pour chaque groupe complet de la file d'attente.

< réponse >

Élément facultatif utilisé pour spécifier la file d'attente de réponses pour les demandes asynchrones.

Attribut	Description
QMGR	Nom du gestionnaire de files d'attente

< tâches >

Élément de groupe devant contenir des éléments qui spécifient les tâches à appeler lorsque les conditions de déclenchement du moniteur sont satisfaites.

< tâche >

Élément de groupe qui définit une tâche individuelle que le moniteur appelle lorsque les conditions de déclenchement sont satisfaites. Actuellement, une seule tâche peut être spécifiée.

< nom >

Nom de la tâche. Accepte tous les caractères alphanumériques.

< description >

Description de la tâche. Toute valeur de texte est autorisée.

< transfer >

Élément de groupe qui définit une tâche de transfert.

< request >

Élément de groupe qui définit le type de tâche. Il doit contenir l'un des éléments suivants qui sont hérités de la définition de schéma `FileTransfer.xsd` :

- [managedTransfer](#)

- managedCall

Attribut	Description
version	Version de la demande fournie par Managed File Transfer. Il s'agit de la forme n.mm où n est la version d'édition principale et mm la version secondaire. Par exemple, 1.00.

< originaire>

Élément de groupe qui contient les éléments spécifiant l'émetteur de la demande.

<hostName>

Nom d'hôte du système sur lequel se trouve le fichier source.

<userID>

ID utilisateur à l'origine du transfert de fichiers.

< IDmqmdUser>

Facultatif. ID utilisateur IBM MQ fourni dans le descripteur de message (MQMD).

< travail>

Élément de groupe contenant des informations sur le travail.

<jobName>

Indique l'identificateur de travail logique.

<defaultVariables>

Élément de groupe contenant un ou plusieurs éléments variable . Ces variables sont utilisées dans la substitution de variables lors de la surveillance d'une file d'attente. Pour plus d'informations sur la substitution de variables, voir «Personnalisation des tâches du moniteur de ressources MFT avec la substitution de variables», à la page 185.

< variable>

Élément contenant la valeur associée à la clé fournie par l'attribut key .

Attribut	Description
clé	Nom de la variable par défaut.

Description du message de suppression du moniteur

Les éléments et attributs utilisés dans les messages du moniteur de suppression sont décrits ci-dessous:

Descriptions des éléments

<deleteMonitor>

Élément de groupe contenant tous les éléments requis pour arrêter et supprimer un moniteur.

Attribut	Description
version	Indique la version de cet élément fournie par Managed File Transfer.

< nom>

Nom du moniteur à supprimer.

< originaire>

Élément de groupe qui contient les éléments spécifiant l'émetteur de la demande.

<hostName>

Nom d'hôte du système sur lequel se trouve le fichier source.

<userID>

ID utilisateur à l'origine du transfert de fichiers.

< IDmqmdUser>

Facultatif. ID utilisateur IBM MQ fourni dans le descripteur de message (MQMD).

< réponse >

Indique le nom de la file d'attente de réponses temporaire générée pour la demande. Le nom de la file d'attente est celui défini par la clé `dynamicQueuePrefix` dans le fichier de configuration `command.properties`. S'il n'est pas spécifié, le nom de la file d'attente a la valeur par défaut `WMQFTE`.

Attribut	Description
QMGR	Nom du gestionnaire de files d'attente de commandes sur lequel la file d'attente dynamique temporaire est générée pour recevoir les réponses.

Exemples

Des exemples de messages XML conformes à ce schéma sont fournis pour chacune des demandes de surveillance suivantes:

- [Créer un moniteur](#)
- [Supprimer un moniteur](#)

Exemples de message de demande de moniteur MFT

Exemples de messages que vous pouvez placer dans la file d'attente de commandes de l'agent pour demander à l'agent de créer ou de supprimer un moniteur.

Demande de création de moniteur

```
<?xml version="1.0" encoding="UTF-8"?>
<monitor:monitor xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xmlns:monitor="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition"
  version="4.00"
  xsi:schemaLocation="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition ./
Monitor.xsd">
  <name>EXAMPLEMONITOR</name>
  <pollInterval>1</pollInterval>
  <agent>US2.BINDINGS.FILE</agent>
  <resources>
    <directory recursionLevel="0">/srv/nfs/incoming</directory>
  </resources>
  <triggerMatch>
    <conditions>
      <allof>
        <condition>
          <fileMatch>
            <pattern>*.completed</pattern>
          </fileMatch>
        </condition>
      </allof>
    </conditions>
  </triggerMatch>
  <reply QMGR="US2.BINDINGS">WMQFTE.4D400F8B20003702</reply>
  <tasks>
    <task>
      <name/>
      <transfer>
        <request xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
          version="4.00"
          xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
          <managedTransfer>
            <originator>
              <hostName>example.com.</hostName>
              <userID>mqm</userID>
            </originator>
            <sourceAgent QMgr="US2.BINDINGS" agent="US2.BINDINGS.FILE"/>
            <destinationAgent QMgr="US2.BINDINGS" agent="US2.BINDINGS.FILE"/>
            <transferSet>
              <item checksumMethod="MD5" mode="binary">
                <source disposition="leave" recursive="false">
                  <file>/srv/nfs/incoming/*.txt</file>
                </source>
                <destination exist="error" type="directory">
                  <file>/srv/backup</file>
                </destination>
              </item>
            </transferSet>
          </managedTransfer>
        </request>
      </transfer>
    </task>
  </tasks>
</monitor:monitor>
```

```

        </transferSet>
      </managedTransfer>
    </request>
  </transfer>
</task>
</tasks>
<originator>
  <hostName>example.com.</hostName>
  <userID>mqm</userID>
</originator>
</monitor:monitor>

```

Demande de suppression de moniteur

```

<?xml version="1.0" encoding="UTF-8"?>
<monitor:deleteMonitor xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xmlns:monitor="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition"
  version="4.00"
  xsi:schemaLocation="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition ./
Monitor.xsd">
  <name>EXAMPLEMONITOR</name>
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
  </originator>
  <reply QMGR="QM_JUPITER">WMQFTE.4D400F8B20003705</reply>
</monitor:deleteMonitor>

```

Référence associée

«[Formats des messages de demande du moniteur MFT](#)», à la page 822

Les moniteurs de ressources sont créés lorsqu'un message XML approprié arrive dans la file d'attente de commandes d'un agent, généralement suite à l'émission par un utilisateur de la commande de surveillance fteCreateou à l'utilisation de l'interface IBM MQ Explorer .

Format de message de demande de l'agent MFT Ping

Vous pouvez envoyer une commande ping à un agent en émettant une commande **ftePingAgent** ou en plaçant un message XML dans la file d'attente de commandes de l'agent. La requête XML de l'agent ping doit être conforme au schéma PingAgent.xsd . Une fois que vous avez installé Managed File Transfer, vous pouvez trouver le fichier schéma PingAgent.xsd dans le répertoire suivant: `MQ_INSTALLATION_PATH/mqft/samples/schema`. Le PingAgent.xsd schéma importe fteutils.xsd, qui se trouve dans le même répertoire.

Lorsque l'agent reçoit un message de demande d'agent ping dans sa file d'attente de commandes, s'il est actif, il renvoie un message de réponse XML à la commande ou à l'application qui place le message de demande d'agent ping dans la file d'attente de commandes. Le message de réponse de l'agent est au format défini par Reply.xsd. Pour plus d'informations sur ce format, voir «[Format du message de réponse de l'agent MFT](#)», à la page 832.

Schéma

Le schéma suivant décrit les éléments valides dans un message XML de demande d'agent ping.

```

<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema"
  xmlns="https://www.ibm.com/xmlns/wmqfte/7.0.1/PingAgent"
  targetNamespace="https://www.ibm.com/xmlns/wmqfte/7.0.1/PingAgent">
  <xsd:include schemaLocation="fteutils.xsd"/>
  <xsd:element name="pingAgent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="originator" type="origRequestType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="agent" type="agentType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="reply" type="replyType" maxOccurs="1" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="version" type="versionType" use="required"/>
    </xsd:complexType>
  </xsd:element>

```

</xsd:schema>

Description du message de demande de l'agent ping

Les éléments et les attributs utilisés dans les messages de demande de l'agent ping sont décrits dans la liste suivante:

<pingAgent>

Élément de groupe contenant tous les éléments requis pour spécifier une demande d'agent ping.

<originaire>

Élément de groupe contenant tous les éléments requis pour spécifier l'émetteur de la demande ping.

<hostName>

Nom d'hôte de la machine sur laquelle la demande a été émise.

<userID>

Nom d'utilisateur de l'émetteur de la demande.

<IDmqmdUser>

Nom d'utilisateur MQMD de l'émetteur de la demande.

<agent>

Agent à envoyer une commande PING.

Attribut	Description
Agent	Obligatoire. Nom de l'agent.
QMgr	Facultatif. Gestionnaire de files d'attente auquel l'agent se connecte.

<réponse>

Nom de la file d'attente à laquelle l'agent doit envoyer le message de réponse.

Attribut	Description
QMGR	Obligatoire. Nom du gestionnaire de files d'attente dans lequel se trouve la file d'attente de réponses.

Exemple

Cet exemple illustre un message d'agent ping envoyé à l'agent AGENT_JUPITER. Si AGENT_JUPITER est actif et peut traiter les demandes d'agent, il envoie un message de réponse à la file d'attente WMQFTE.4D400F8B20003708 sur QM_JUPITER.

```
<?xml version="1.0" encoding="UTF-8"?>
<ping:pingAgent xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xmlns:ping="https://www.ibm.com/xmlns/wmqfte/7.0.1/PingAgent"
  version="4.00">
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
  </originator>
  <agent agent="AGENT_JUPITER" QMgr="QM_JUPITER"/>
  <reply QMGR="QM_JUPITER">WMQFTE.4D400F8B20003708</reply>
</ping:pingAgent>
```

Format du message de réponse de l'agent MFT

Lorsqu'un agent reçoit un message XML dans sa file d'attente de commandes d'agent, si une réponse est requise, l'agent envoie un message de réponse XML à la file d'attente de réponses définie dans le message d'origine. Le code XML de réponse est conforme au schéma Reply.xsd. Le document de schéma Reply.xsd se trouve dans le répertoire *MQ_INSTALLATION_PATH/mqft/samples/schema*. Le Reply.xsd schéma importe fteutils.xsd, qui se trouve dans le même répertoire.

Formats de message MFT pour la sécurité

Cette rubrique décrit les messages publiés dans le gestionnaire de files d'attente de coordination Managed File Transfer en rapport avec la sécurité.

Message de journal non autorisé

Si la vérification des droits utilisateur est activée, l'agent peut publier des messages non autorisés dans le gestionnaire de files d'attente de coordination. «Restriction des droits utilisateur sur les actions de l'agent MFT», à la page 368 décrit comment activer la vérification des droits utilisateur.

Chaque fois qu'un utilisateur soumet une demande d'exécution d'une action restreinte à l'agent, à l'aide d'une commande Managed File Transfer ou du plug-in IBM MQ Explorer, l'agent vérifie que l'utilisateur est autorisé à effectuer l'action. Si l'utilisateur ne parvient pas à vérifier les droits d'accès, un message de journal non autorisé est publié dans le gestionnaire de files d'attente de coordination sur son système SYSTEM.FTE/Log/nom_agent/NotAuthorized .

Ce message est conforme au schéma XML TransferLog.xsd . Exemple :

```
<?xml version="1.0" encoding="UTF-8"?>
<notAuthorized version="3.00"
  ID="414d5120716d312020202020202020204da5924a2010ce03"
  agentRole="sourceAgent"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="TransferLog.xsd"
  xmlns="">
  <action time="2009-08-28T12:31:15.781Z">not_authorized</action>
  <originator>
    <mqmdUserID>test1</mqmdUserID>
  </originator>
  <authority>administration</authority>
  <status resultCode="53">
    <supplement>BFGCH0083E: The user (test1) does not have the authority (ADMINISTRATION) required
to shut down agent 'AGENT'.</supplement>
    <supplement>
&lt;?xml version=&quot;1.0&quot; encoding=&quot;UTF-8&quot;?&gt;
&lt;internal:request version=&quot;3.00&quot; xmlns:xsi=&quot;https://www.w3.org/2001/XMLSchema-
instance&quot;
  xmlns:internal=&quot;http://wmqfte.ibm.com/internal&quot;&gt;
&lt;internal:shutdown_agent=&quot;SYSTEM.FTE.COMMAND.AGENT&quot; hostname= &quot;qm1&quot;
mode=&quot;controlled&quot;/&gt;
&lt;reply QMGR=&quot;qm1&quot;&gt;WMQFTE.4A92A54D02CE1020&lt;/reply&gt;
&lt;/internal:request&gt;
    </supplement>
  </status>
</notAuthorized>
```

Ce message est un journal contenant les informations suivantes:

- A l'origine de la demande
- Niveau de droit d'accès Managed File Transfer requis pour traiter la demande
- Statut de la demande
- Spécification de la demande

Description du message de journal non autorisé

Les éléments et attributs utilisés dans le message non autorisé sont décrits ci-dessous:

<notAuthorized>

Élément de groupe qui décrit une vérification d'autorisation d'utilisateur ayant échoué.

Attribut	Description
version	Indique la version de cet élément, telle que détaillée par Managed File Transfer.
ID	Identificateur unique de la demande qui n'a pas été autorisée.

< originaire>

Élément de groupe qui contient les éléments spécifiant l'émetteur de la demande.

< autorité>

Indique le niveau de droits d'accès Managed File Transfer dont l'utilisateur a besoin pour effectuer l'action demandée.

< IDmqmdUser>

ID utilisateur IBM MQ fourni dans le descripteur de message (MQMD)

<action>

Indique le statut d'autorisation de la demande correspondant à l'attribut ID de l'élément <notAuthorized>.

Attribut	Description
heure	Indique la date et l'heure de publication de l'entrée de journal (au format date / heure).

< statut>

Code de résultat et messages complémentaires.

Format de fichier de données d'identification MFT

Le fichier MQMFTCcredentials.xml contient les informations d'ID utilisateur et de mot de passe sensibles. Les éléments du fichier MQMFTCcredentials.xml doivent être conformes au schéma MQMFTCcredentials.xsd. La sécurité des fichiers de données d'identification est de la responsabilité de l'utilisateur.

V 9.0.3 ➔ **V 9.0.0.2** Depuis IBM MQ 9.0.0 Fix Pack 2 et IBM MQ 9.0.3, vous pouvez désactiver le mode de compatibilité par défaut et activer l'authentification MQCSP pour un agent Managed File Transfer en ajoutant un nouveau paramètre, **useMQCSPAauthentication**, au MFT fichier de données d'identification MQMFTCcredentials.xml de l'utilisateur concerné et en le définissant sur true. Pour plus d'informations, voir la rubrique relative à [l'activation du mode d'authentification MQCSP](#).

➔ **z/OS** ➔ **z/OS** À partir de IBM MQ 8.0, le fichier MQMFTCcredentials.xsd peut également être un membre PDSE sous z/OS.

Avant IBM WebSphere MQ 7.5, les informations contenues dans le fichier MQMFTCcredentials.xml étaient stockées dans des fichiers de propriétés distincts.

Le fichier MQMFTCcredentials.xml doit être conforme au schéma MQMFTCcredentials.xsd. Le document de schéma MQMFTCcredentials.xml se trouve dans le répertoire MQ_INSTALLATION_PATH/mqft/samples/schema de l'installation Managed File Transfer.

Schéma

Le schéma suivant décrit les éléments valides dans le fichier MQMFTCcredentials.xml.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  @start_non_restricted_prolog@
  Version: %Z% %I% %W% %E% %U% [%H% %T%]

  Licensed Materials - Property of IBM

  5724-H72

  Copyright IBM Corp. 2012, 2023. All Rights Reserved.

  US Government Users Restricted Rights - Use, duplication or
  disclosure restricted by GSA ADP Schedule Contract with
  IBM Corp.
  @end_non_restricted_prolog@
-->

<!--
  This schema defines the format of an MQMFTCcredentials file. Files of this type
```

```

store credential information for agent and logger processes. They can contain
user names and passwords either in clear text or which have been obfuscated
using the fteObfuscate command.
-->

<!-- Example mqmftCredentials.xml file:
<?xml version="1.0" encoding="UTF-8"?>
<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:logger name="LOG1" user="user1" password="passw0rd"/>
  <tns:logger name="ORACLE" userCipher="kj2h3dfkgf" passwordCipher="1a3n67eae" />
  <tns:file path="/home/emma/trust.jks" password="passw0rd"/>
  <tns:file path="/var/tmp/keystore.jks" passwordCipher="e71vKCg2pf" />

  <tns:qmgr name="QM_COORD" user="tim" mqUserId="user1" mqPassword="passw0rd"/>
  <tns:qmgr name="QM_COORD" user="tom" mqUserId="user1" mqPasswordCipher="e71vKCg2pf"/>
V 9.0.3 <tns:qmgr name="QM_COORD" user="ernest" mqUserId="ernest"
mqPassword="AveryL0ngPassw0rd2135" useMQCSPAAuthentication="true"/>
</tns:mqmftCredentials>
-->

<schema targetNamespace="http://wmqfte.ibm.com/MQMFTCredentials"
  elementFormDefault="qualified"
  xmlns="https://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://wmqfte.ibm.com/MQMFTCredentials">

  <element name="mqmftCredentials" type="tns:mqmftCredentialsType"/>

  <complexType name="mqmftCredentialsType">
    <sequence>
      <choice minOccurs="0" maxOccurs="unbounded">
        <element name="logger" type="tns:loggerType"/>
        <element name="file" type="tns:fileType"/>
        <element name="qmgr" type="tns:mqUserPassType"/>
      </choice>
    </sequence>
  </complexType>

  <complexType name="loggerType">
    <attribute name="name" type="string" use="required"/>
    <attribute name="user" type="string" use="optional"/>
    <attribute name="userCipher" type="string" use="optional"/>
    <attribute name="password" type="string" use="optional"/>
    <attribute name="passwordCipher" type="string" use="optional"/>
  </complexType>

  <complexType name="fileType">
    <attribute name="path" type="string" use="required"/>
    <attribute name="password" type="string" use="optional"/>
    <attribute name="passwordCipher" type="string" use="optional"/>
  </complexType>

  <!-- Example XML:

  <tns:qmgr name="QM_COORD" user="tim" mqUserId="user1" mqPassword="passw0rd"/>
  <tns:qmgr name="QM_COORD" user="tom" mqUserIdCipher="xh5U7812x"
mqPasswordCipher="e71vKCg2pf"/>
  <tns:qmgr name="QM_COORD" mqUserId="defaultUser" mqPassword="passw0rd"/>
V 9.0.3 <tns:qmgr name="QM_COORD" user="ernest" mqUserId="ernest"
mqPassword="AveryL0ngPassw0rd2135" useMQCSPAAuthentication="true"/>
-->

  <complexType name="mqUserPassType">
    <attribute name="name" type="string" use="required"/>
    <attribute name="user" type="string" use="optional"/>
    <attribute name="mqUserId" type="string" use="optional"/>
    <attribute name="mqUserIdCipher" type="string" use="optional"/>
    <attribute name="mqPassword" type="string" use="optional"/>
    <attribute name="mqPasswordCipher" type="string" use="optional"/>
V 9.0.3 <attribute name="useMQCSPAAuthentication" type="boolean" use="optional"/>
  </complexType>

</schema>

```

Description du fichier MQMFTCredentials.xml

Les éléments et attributs utilisés dans le fichier MQMFTCredentials.xml sont décrits dans la liste suivante.

<mqmftCredentials>

Élément racine du document XML.

<fichier>

Fichier dans le transfert.

Attribut	Description
chemin	Chemin d'accès au fichier de clés ou de clés certifiées en cours d'accès.
mot de passe	Mot de passe permettant d'accéder au fichier.

<logger>

Consignateur responsable de l'activité de consignation.

Attribut	Description
nom	Nom du consignateur.
utilisateur	Nom d'utilisateur que le consignateur utilisera pour se connecter à sa base de données.
mot de passe	Mot de passe utilisé par le consignateur pour se connecter à sa base de données.

< gestionnaire_files_attente >

Connexion du gestionnaire de files d'attente IBM MQ .

Attribut	Description
nom	Nom du gestionnaire de files d'attente IBM MQ associé.
utilisateur	Facultatif: nom de l'utilisateur demandant la connexion.
ID mqUserou mqUserIdCipher	ID utilisateur de texte en clair (mqUserId) ou ID utilisateur de texte brouillé (mqUserIdCipher) à fournir à un gestionnaire de files d'attente IBM MQ .
Chiffrement mqPassword ou mqPassword	Mot de passe en texte clair (mqPassword) ou mot de passe en texte brouillé (chiffrementmqPassword) à fournir à un gestionnaire de files d'attente IBM MQ .

Remarque : Le fichier MQMFTCredentials.xml peut contenir des informations sensibles. Par conséquent, lorsqu'il est créé, vérifiez que les droits d'accès au fichier sont vérifiés. Lorsque vous utilisez un bac à sable, définissez-le comme étant exclu. Pour plus d'informations sur les bacs à sable, voir «Utilisation des bacs à sable d'agent MFT», à la page 46.

Concepts associés

«Configuration de MQMFTCredentials.xml sur z/OS», à la page 71

Si Managed File Transfer (MFT) est configuré avec la sécurité activée, l'authentification de connexion requiert tous les agents MFT , ainsi que les commandes qui se connectent à un gestionnaire de files d'attente, pour fournir les données d'identification par ID utilisateur et mot de passe.

Référence associée

«Authentification de connexion MFT et IBM MQ», à la page 42

L'authentification de connexion permet à un gestionnaire de files d'attente d'être configuré pour authentifier les applications à l'aide d'un ID utilisateur et d'un mot de passe fournis. Si la sécurité est activée pour le gestionnaire de files d'attente associé et que les données d'identification (ID utilisateur et mot de passe) sont requises, la fonction d'authentification de connexion doit être activée pour que la

connexion à un gestionnaire de files d'attente puisse être établie. L'authentification de connexion peut être exécutée en mode compatibilité ou en mode d'authentification MQCSP.

«[fteObfuscate: chiffrement des données sensibles](#)», à la page 507

La commande **fteObfuscate** chiffre les données sensibles dans les fichiers de données d'identification. Ainsi, le contenu des fichiers de données d'identification n'est plus lu par une personne qui a accès au fichier.

Format de fichier des données d'identification du pont de protocole

Le fichier `ProtocolBridgeCredentials.xml` du répertoire de configuration Managed File Transfer Agent définit les noms d'utilisateur et les données d'identification que l'agent de pont de protocole utilise pour s'autoriser avec le serveur de protocole.

Le fichier `ProtocolBridgeCredentials.xml` doit être conforme au schéma `ProtocolBridgeCredentials.xsd`. Le document de schéma `ProtocolBridgeCredentials.xsd` se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/schema` de l'installation MQMFT. Les utilisateurs sont responsables de la création manuelle du fichier `ProtocolBridgeCredentials.xml`, qui n'est plus créé par la commande **fteCreateBridgeAgent**. Des exemples de fichier sont disponibles dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples` de l'installation MQMFT.

IBM WebSphere MQ 7.5 a introduit un nouvel élément `<agent>` qui contient l'élément `<server>` ou `<serverHost>` pour l'agent nommé.

Le fichier `ProtocolBridgeCredentials.xml` est rechargé périodiquement par l'agent et toute modification valide apportée au fichier aura une incidence sur le comportement de l'agent. L'intervalle de rechargement par défaut est de 30 secondes. Cet intervalle peut être modifié en spécifiant la propriété d'agent `xmlConfigReloadInterval` dans le fichier `agent.properties`.

Schéma- 7.5 ou version ultérieure

Le schéma suivant décrit les éléments valides dans le fichier `ProtocolBridgeCredentials.xml` pour IBM MQ 8.

```
<schema targetNamespace="http://wmqfte.ibm.com/ProtocolBridgeCredentials" elementFormDefault="qualified"
  xmlns="https://www.w3.org/2001/XMLSchema" xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeCredentials">
  <!--
  <?xml version="1.0" encoding="UTF-8"?>
  <tns:credentials xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeCredentials"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeCredentials
  ProtocolBridgeCredentials.xsd ">
    <tns:agent name="agent1">
      <tns:serverHost name="myserver">
        <tns:user name="fred" serverPassword="pwd" serverUserId="bill"/>
        <tns:user name="jane" serverUserId="june" hostKey="1F:2e:f3">
          <tns:privateKey associationName="test" keyPassword="pwd2">
            .... private key ...
          </tns:privateKey>
        </tns:user>
      </tns:serverHost>
    </tns:agent>

    <tns:agent name="agent2">
      <tns:server name="server*" pattern="wildcard">
        <tns:user name="fred" serverPassword="pwd" serverUserId="bill"/>
        <tns:user name="jane" serverUserId="june" hostKey="1F:2e:f3">
          <tns:privateKey associationName="test" keyPassword="pwd2">
            .... private key ...
          </tns:privateKey>
        </tns:user>
      </tns:server>
    </tns:agent>

    <tns:agent name="agent3">
      <tns:serverHost name="ftpsServer"
        keyStorePassword="keypass"
        trustStorePassword="trustpass">
```

```

        <tns:user name="fred" serverPassword="pwd" serverUserId="bill" />
    </tns:serverHost>
</tns:agent>

</tns:credentials>
-->

<element name="credentials" type="tns:credentialsType" />

<complexType name="credentialsType">
    <sequence>
        <element name="agent" type="tns:agentType" minOccurs="0" maxOccurs="unbounded" />
    </sequence>
</complexType>

<complexType name="agentType">
    <choice minOccurs="0" maxOccurs="1">
        <element name="serverHost" type="tns:serverHostType" minOccurs="0" maxOccurs="unbounded" />
        <element name="server" type="tns:serverType" minOccurs="0" maxOccurs="unbounded" />
    </choice>
    <attribute name="name" type="string" use="required" />
</complexType>

<complexType name="serverHostType">
    <sequence>
        <element ref="tns:user" minOccurs="0" maxOccurs="unbounded" />
    </sequence>
    <attribute name="name" type="string" use="required" />
    <attribute name="keyStorePassword" type="string" use="optional" />
    <attribute name="keyStorePasswordCipher" type="string" use="optional" />
    <attribute name="trustStorePassword" type="string" use="optional" />
    <attribute name="trustStorePasswordCipher" type="string" use="optional" />
</complexType>

<complexType name="serverType">
    <sequence>
        <element ref="tns:user" minOccurs="0" maxOccurs="unbounded" />
    </sequence>
    <attribute name="name" type="string" use="required" />
    <attribute name="pattern" type="tns:patternType" use="optional" />
    <attribute name="keyStorePassword" type="string" use="optional" />
    <attribute name="keyStorePasswordCipher" type="string" use="optional" />
    <attribute name="trustStorePassword" type="string" use="optional" />
    <attribute name="trustStorePasswordCipher" type="string" use="optional" />
</complexType>

<element name="user" type="tns:userType" />

<complexType name="userType">
    <sequence>
        <element ref="tns:privateKey" minOccurs="0" maxOccurs="unbounded" />
    </sequence>
    <attribute name="name" type="string" use="required" />
    <attribute name="serverUserId" type="string" use="optional" />
    <attribute name="serverUserIdCipher" type="string" use="optional" />
    <attribute name="serverPassword" type="string" use="optional" />
    <attribute name="serverPasswordCipher" type="string" use="optional" />
    <attribute name="hostKey" use="optional">
        <simpleType>
            <restriction base="string">
                <pattern
                    value="([a-zA-F0-9]){2}(:([a-zA-F0-9]){2})*">
                </pattern>
            </restriction>
        </simpleType>
    </attribute>
</complexType>

<element name="privateKey" type="tns:privateKeyType" />

<complexType name="privateKeyType">
    <simpleContent>
        <extension base="string">
            <attribute name="keyPassword" type="string" use="optional" />
            <attribute name="keyPasswordCipher" type="string" use="optional" />
            <attribute name="associationName" type="string" use="required" />
        </extension>
    </simpleContent>
</complexType>

<!--
    Determines the type of pattern matching to use.

```

```

-->
<simpleType name="patternType">
  <restriction base="string">
    <enumeration value="regex"/>
    <enumeration value="wildcard"/>
  </restriction>
</simpleType>
</schema>

```

Description du fichier ProtocolBridgeCredentials.xml

Les éléments et attributs utilisés dans le fichier ProtocolBridgeCredentials.xml sont décrits dans la liste suivante.

< données_d'identification >

Élément de groupe contenant des éléments qui décrivent les données d'identification utilisées par un agent de pont de protocole pour se connecter à un serveur de protocole.

< agent >

Élément contenant une définition < server > ou < serverHost > pour un agent nommé.

< server >

Serveur de protocole auquel le pont de protocole se connecte.

L'élément < server > n'est pas pris en charge pour 7.0.4 ou version antérieure.

Attribut	Description
nom	Nom du serveur de protocole.
schéma	Si vous avez utilisé des caractères génériques ou des expressions régulières pour spécifier le modèle d'un nom de serveur de protocole, utilisez wildcard ou regex.
trustStoreMot de passe ou trustStorePasswordCipher	Obligatoire lorsque l'élément < serveur > fait référence à un serveur FTPS. Mot de passe utilisé pour accéder au magasin de clés de confiance. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
keyStoreMot de passe ou keyStorePasswordCipher	Facultatif. Mot de passe utilisé pour accéder au magasin de clés. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.

< serverHost >

Nom d'hôte du serveur de protocole auquel le pont de protocole se connecte.

Le fichier ProtocolBridgeCredentials.xml peut contenir des éléments < serverHost > ou des éléments < server >, mais vous ne pouvez pas utiliser une combinaison des deux types différents.

Lorsque vous utilisez < serverHost >, le nom est comparé au nom d'hôte du serveur de protocole.

Lorsque vous utilisez < server >, le nom est comparé au nom du serveur de protocole (tel que défini dans le fichier ProtocolBridgeProperties.xml).

Attribut	Description
nom	Nom d'hôte ou adresse IP du serveur de protocole.
trustStoreMot de passe ou trustStorePasswordCipher	Obligatoire lorsque l'élément < serverHost > fait référence à un serveur FTPS. Mot de passe utilisé pour accéder au magasin de clés de confiance. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.

Attribut	Description
keyStoreMot de passe ou keyStorePasswordCipher	Facultatif. Mot de passe utilisé pour accéder au magasin de clés. Cette propriété est facultative sauf si vous définissez l'attribut keyStore , auquel cas elle est obligatoire. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.

< utilisateur >

Mappage d'un nom d'utilisateur Managed File Transfer à un nom d'utilisateur de serveur de protocole.

Attribut	Description
nom	Nom d'utilisateur utilisé avec Managed File Transfer.
serverUserID ou serverUserIdCipher	Nom d'utilisateur utilisé avec le serveur de protocole. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
Chiffrement serverPassword ou serverPassword	Mot de passe du nom d'utilisateur utilisé sur le serveur de protocole. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
hostKey	Empreinte digitale SSH de l'hôte du serveur.

<privateKey >

Clé privée d'un utilisateur.

Attribut	Description
keyPassword ou keyStorePasswordCipher	Mot de passe de la clé privée. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
associationName	Nom utilisé pour la trace et la consignation.

Format du fichier de propriétés du pont de protocole

Le fichier ProtocolBridgeProperties.xml du répertoire de configuration de l'agent définit les propriétés des serveurs de fichiers de protocole.

Le fichier ProtocolBridgeProperties.xml doit être conforme au schéma ProtocolBridgeProperties.xsd . Le document de schéma ProtocolBridgeProperties.xsd se trouve dans le répertoire MQ_INSTALLATION_PATH/mqft/samples/schema de l'installation Managed File Transfer . Un fichier modèle, ProtocolBridgeProperties.xml, est créé par la commande **fteCreateBridgeAgent** dans le répertoire de configuration de l'agent.

Le fichier ProtocolBridgeProperties.xml est rechargé périodiquement par l'agent et toute modification valide apportée au fichier aura une incidence sur le comportement de l'agent. L'intervalle de rechargement par défaut est de 30 secondes. Vous pouvez modifier cet intervalle en spécifiant la propriété d'agent xmlConfigReloadInterval dans le fichier agent.properties .

Schéma

Le schéma suivant décrit le fichier ProtocolBridgeProperties.xml .

Remarque : Les attributs maxReconnectRetry et reconnectWaitPeriod ne sont pas pris en charge sous IBM WebSphere MQ 7.5, IBM WebSphere MQ File Transfer Edition 7.0.2ou version ultérieure.

```
<schema targetNamespace="http://wmqfte.ibm.com/ProtocolBridgeProperties" elementFormDefault="qualified"
  xmlns="https://www.w3.org/2001/XMLSchema" xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeProperties">
  <!--
  Example: ProtocolBridgeProperties.xml
```

```

<?xml version="1.0" encoding="UTF-8"?>
<tns:serverProperties xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeProperties"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeProperties
  ProtocolBridgeProperties.xsd">
  <tns:credentialsFile path="$HOME/ProtocolBridgeCredentials.xml"/>
  <tns:defaultServer name="myserver"/>
  <tns:ftpServer name="myserver" host="myhost.hursley.ibm.com" port="1234" platform="windows"
    timeZone="Europe/London" locale="en-GB" fileEncoding="UTF-8"
    listFormat="unix" limitedWrite="false"/>
  <tns:sftpServer name="server1" host="myhost.hursley.ibm.com" platform="windows"
    fileEncoding="UTF-8" limitedWrite="false">
    <limits maxListFileNames="10"/>
  </tns:sftpServer>
</tns:serverProperties>
-->

<!-- Root element for the document -->
<element name="serverProperties" type="tns:serverPropertiesType"></element>

<!--
  A container for all protocol bridge server properties
-->
<complexType name="serverPropertiesType">
  <sequence>
    <element name="credentialsFile" type="tns:credentialsFileName" minOccurs="0" maxOccurs="1"/>
    <element name="defaultServer" type="tns:serverName" minOccurs="0" maxOccurs="1"/>
    <choice minOccurs="0" maxOccurs="unbounded">
      <element name="ftpServer" type="tns:ftpServerType"/>
      <element name="sftpServer" type="tns:sftpServerType"/>
      <element name="ftpsServer" type="tns:ftpsServerType"/>
      <element name="ftpsfgServer" type="tns:ftpsfgServerType"/>
      <element name="ftpsfgServer" type="tns:ftpsfgServerType"/>
    </choice>
  </sequence>
</complexType>

<!--
  A container for a server name
-->
<complexType name="serverName">
  <attribute name="name" type="tns:serverNameType" use="required"/>
</complexType>

<!--
  A container for a credentials file name
-->
<complexType name="credentialsFileName">
  <attribute name="path" type="string" use="required"/>
</complexType>

<!--
  A container for all the information about an FTP server
-->
<complexType name="ftpServerType">
  <sequence>
    <element name="limits" type="tns:generalLimitsType" minOccurs="0" maxOccurs="1" />
  </sequence>
  <attributeGroup ref="tns:ftpServerAttributes"/>
  <attribute name="passiveMode" type="boolean" use="optional"/>
</complexType>

<!--
  A container for all the information about an SFG FTP server
-->
<complexType name="ftpsfgServerType">
  <sequence>
    <element name="limits" type="tns:generalLimitsType" minOccurs="0" maxOccurs="1" />
  </sequence>
  <attributeGroup ref="tns:ftpServerAttributes"/>
</complexType>

<!--
  A container for all the information about an SFTP server
-->
<complexType name="sftpServerType">
  <sequence>
    <element name="limits" type="tns:sftpLimitsType" minOccurs="0" maxOccurs="1" />
  </sequence>
  <attributeGroup ref="tns:sftpServerAttributes"/>
</complexType>

```

```

<!--
  A container for all the information about a FTPS server
-->
<complexType name="ftpsServerType">
  <sequence>
    <element name="limits" type="tns:generalLimitsType" minOccurs="0" maxOccurs="1" />
  </sequence>
  <attributeGroup ref="tns:ftpsServerAttributes"/>
</complexType>

<!--
  A container for all the information about a SFG FTPS server
-->
<complexType name="ftpsSfgServerType">
  <sequence>
    <element name="limits" type="tns:generalLimitsType" minOccurs="0" maxOccurs="1" />
  </sequence>
  <attributeGroup ref="tns:ftpsServerAttributes"/>
</complexType>

<!--
  Attributes common to all server types
-->
<attributeGroup name="generalServerAttributes">
  <attribute name="name" type="tns:serverNameType" use="required"/>
  <attribute name="host" type="string" use="required"/>
  <attribute name="port" type="nonNegativeInteger" use="optional"/>
  <attribute name="platform" type="tns:platformType" use="required"/>
  <attribute name="fileEncoding" type="string" use="required"/>
  <attribute name="limitedWrite" type="boolean" use="optional"/>
  <attribute name="controlEncoding" type="string" use="optional"/>
</attributeGroup>

<!--
  Attributes common to ftp and ftps server types
-->
<attributeGroup name="ftpServerAttributes">
  <attributeGroup ref="tns:generalServerAttributes"/>
  <attribute name="timeZone" type="string" use="required"/>
  <attribute name="locale" type="tns:localeType" use="required"/>
  <attribute name="listFormat" type="tns:listFormatType" use="optional"/>
  <attribute name="listFileRecentDateFormat" type="tns:dateFormatType" use="optional"/>
  <attribute name="listFileOldDateFormat" type="tns:dateFormatType" use="optional"/>
  <attribute name="monthShortNames" type="tns:monthShortNamesType" use="optional"/>
</attributeGroup>

<!--
  Attributes common to ftps server types
-->
<attributeGroup name="ftpsServerAttributes">
  <attributeGroup ref="tns:ftpServerAttributes"/>
  <attribute name="ftpsType" type="tns:ftpsTypeType" use="optional"/>
  <attribute name="trustStore" type="string" use="required"/>
  <attribute name="trustStoreType" type="string" use="optional"/>
  <attribute name="keyStore" type="string" use="optional"/>
  <attribute name="keyStoreType" type="string" use="optional"/>
  <attribute name="ccc" type="boolean" use="optional"/>
  <attribute name="protFirst" type="boolean" use="optional"/>
  <attribute name="auth" type="string" use="optional"/>
  <attribute name="connectTimeout" type="nonNegativeInteger" use="optional"/>
</attributeGroup>

<!--
  A container for limit-type attributes for a server. Limit parameters
  are optional, and if not specified a system default will be used.
-->
<complexType name="generalLimitsType">
  <attributeGroup ref="tns:generalLimitAttributes"/>
</complexType>

<complexType name="sftplimitsType">
  <attributeGroup ref="tns:generalLimitAttributes"/>
  <attribute name="connectionTimeout" type="nonNegativeInteger" use="optional"/>
</complexType>

<!--
  Attributes for limits common to all server types
-->
<attributeGroup name="generalLimitAttributes">

```

```

    <attribute name="maxListFileNames" type="positiveInteger" use="optional"/>
    <attribute name="maxListDirectoryLevels" type="nonNegativeInteger" use="optional"/>
    <attribute name="maxReconnectRetry" type="nonNegativeInteger" use="optional"/>
    <attribute name="reconnectWaitPeriod" type="nonNegativeInteger" use="optional"/>
    <attribute name="maxSessions" type="positiveInteger" use="optional"/>
    <attribute name="socketTimeout" type="nonNegativeInteger" use="optional"/>
</attributeGroup>

<!--
    The type for matching valid server names. Server names must be at least 2 characters in length
and
    are limited to alphanumeric characters and the following characters: ".", "_", "/" and "%".
-->
<simpleType name="serverNameType">
    <restriction base="string">
        <pattern value="[0-9a-zA-Z\._/%]{2,}" />
    </restriction>
</simpleType>

<!--
    The types of platform supported.
-->
<simpleType name="platformType">
    <restriction base="string">
    </restriction>
</simpleType>

<!--
    The type for matching a locale specification.
-->
<simpleType name="localeType">
    <restriction base="string">
        <pattern value="(.)[-_](.)/" />
    </restriction>
</simpleType>

<!--
    The types of list format supported (for FTP servers).
-->
<simpleType name="listFormatType">
    <restriction base="string">
    </restriction>
</simpleType>

<!--
    Date format for FTP client directory listing on an FTP server. This is
    the format to be passed to methods setDefaultDateFormatStr and
    setRecentDateFormatStr for Java class:
    org.apache.commons.net.ftp.FTPClientConfig
-->
<simpleType name="dateFormatType">
    <restriction base="string">
    </restriction>
</simpleType>

<!--
    A list of language-defined short month names can be specified. These are
    used for translating the directory listing received from the FTP server.
    The format is a string of three character month names separated by "|"
-->
<simpleType name="monthShortNamesType">
    <restriction base="string">
        <pattern value="(...\|){11}(...)" />
    </restriction>
</simpleType>

<!--
    The enumerations of the allowed FTPS types: "implicit" & "explicit"
    If not specified the default is "explicit"
-->
<simpleType name="ftpsTypeType">
    <restriction base="string">
        <enumeration value="explicit"/>
        <enumeration value="implicit"/>
    </restriction>
</simpleType>

<!--
    Attribute Group for SFTP Servers
-->
<attributeGroup name="sftpServerAttributes">
    <attributeGroup ref="tns:generalServerAttributes"/>

```

```
<attribute name="cipherList" type="string" use="optional"/>
</attributeGroup>
</schema>
```

Description du fichier ProtocolBridgeProperties.xml

Les éléments et attributs utilisés dans le fichier ProtocolBridgeProperties.xml sont décrits dans la liste suivante:

<serverProperties>

Élément racine du document XML

<credentialsFile>

Chemin d'accès au fichier contenant les données d'identification. Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir [«Utilisation des variables d'environnement dans les propriétés MFT»](#), à la page 547

<defaultServer>

Serveur de fichiers de protocole qui fait office de serveur par défaut pour les transferts de fichiers

<ftpServer>

Un serveur de fichiers FTP

<sftpServer>

Un serveur de fichiers SFTP

<ftpsServer>

Un serveur de fichiers FTPS

Attributs de serveur généraux qui s'appliquent à tous les types de serveur de fichiers de protocole:

Attribut	Description
nom	Obligatoire. Nom du serveur de fichiers de protocole. Les noms de serveur de protocole doivent comporter au moins deux caractères, ne sont pas sensibles à la casse et sont limités aux caractères alphanumériques et aux caractères suivants: <ul style="list-style-type: none"> • Point (.) • trait de soulignement (_) • barre oblique (/) • signe pourcentage (%)
host	Obligatoire. Nom d'hôte ou adresse IP du serveur de fichiers de protocole à partir duquel vous souhaitez envoyer ou recevoir des fichiers.
port	Facultatif. Numéro de port du serveur de fichiers de protocole à partir duquel vous souhaitez envoyer ou recevoir des fichiers.
plateforme	Obligatoire. Plateforme du serveur de fichiers de protocole à partir de laquelle vous souhaitez envoyer ou recevoir des fichiers. Indiquez UNIX ou WINDOWS. Définissez cette propriété en fonction de la manière dont vous entrez les chemins d'accès sur votre serveur FTP, FTPS ou SFTP. Par exemple, si vous exécutez un serveur FTP sous Windows, mais que vous vous connectez au serveur, vous devez entrer des chemins de style UNIX(c'est-à-dire avec des barres obliques), définir cette valeur sur UNIX et non sur WINDOWS. Les serveurs s'exécutant sous Windows présentent souvent un système de fichiers de style UNIX.
fileEncoding	Obligatoire. Définit le codage de caractères utilisé par le serveur de fichiers. Cette propriété est utilisée lorsque vous transférez des fichiers en mode texte afin que les séquences de codage correctes soient modifiées lorsque les fichiers sont déplacés entre les plateformes. Par exemple, UTF-8.

Attribut	Description
limitedWrite	Facultatif. Le mode par défaut lors de l'écriture sur un serveur de fichiers consiste à créer un fichier temporaire, puis à renommer ce fichier lorsque le transfert est terminé. Pour un serveur de fichiers configuré en écriture seule, le fichier est créé directement avec son nom final. La valeur de cette propriété peut être true ou false. La valeur par défaut est false.
controlEncoding	Facultatif. Valeur de codage de contrôle pour les messages de contrôle envoyés au serveur de fichiers de protocole. Cette propriété affecte le codage du nom de fichier utilisé et doit être compatible avec le codage de contrôle du serveur de fichiers de protocole. Valeur par défaut : UTF-8.

Attributs généraux qui s'appliquent uniquement aux serveurs FTP et FTPS:

Attribut	Description
timeZone	Obligatoire. Fuseau horaire du serveur de fichiers de protocole à partir duquel vous souhaitez envoyer ou recevoir des fichiers. Par exemple: America / New_York ou Asia / Tokyo.
paramètres_régionaux	Obligatoire. Langue utilisée sur le serveur de fichiers de protocole pour l'envoi ou la réception de fichiers. Par exemple: en_US ou ja_JP
listFormat	Facultatif. Format de liste qui définit le format des informations de liste de fichiers renvoyées par le serveur de fichiers de protocole. Utilisez Windows ou UNIX. La valeur par défaut est UNIX.
Format listFileRecentDate	Facultatif. Format de date récent (moins d'un an) pour la liste des répertoires de client FTP sur un serveur FTP. Cet attribut et l'attribut de format listFileOldDate vous permettent de redéfinir les formats de date attendus renvoyés par le serveur de fichiers de protocole. La valeur par défaut est celle définie par le serveur de fichiers de protocole.
Format listFileOldDate	Facultatif. Ancien format de date (plus d'un an) pour la liste des répertoires de client FTP sur un serveur FTP. Cet attribut et l'attribut de format listFileRecentDate vous permettent de redéfinir les formats de date attendus renvoyés par le serveur de fichiers de protocole. La valeur par défaut est celle définie par le serveur de fichiers de protocole.
Noms monthShort	Facultatif. Liste de remplacement des noms de mois utilisés pour décoder les informations de date renvoyées par le serveur de fichiers de protocole. Cette propriété est constituée d'une liste de 12 noms, séparés par des virgules, pour remplacer les valeurs de mois de l'environnement local par défaut. La valeur par défaut est celle définie par le serveur de fichiers de protocole.

Attributs généraux qui s'appliquent uniquement aux serveurs FTP:

Attribut	Description
passiveMode	Facultatif. Contrôle si la connexion au serveur FTP est passive ou active. Si vous définissez la valeur de cette propriété sur false, la connexion est active. Si vous définissez la valeur sur true, la connexion est passive. La valeur par défaut est false.

Attributs généraux qui s'appliquent uniquement aux serveurs FTPS:

Attribut	Description
ftpsType	Facultatif. Indique si la forme explicite ou implicite du protocole FTPS est utilisée. La valeur par défaut est <code>explicit</code> .
trustStore	Obligatoire. Emplacement du magasin de clés de confiance utilisé pour déterminer si le certificat présenté par le serveur FTPS est digne de confiance.
trustStoreType	Facultatif. Format du fichier de clés certifiées. La valeur par défaut est <code>JKS</code> .
keyStore	Facultatif. Emplacement du magasin de clés utilisé pour fournir des informations de certificat en cas d'authentification par le serveur FTPS. Par défaut, le pont de protocole ne peut pas se connecter aux serveurs FTPS configurés pour exiger l'authentification des clients.
KeyStoreType	Facultatif. Format du fichier de clés. La valeur par défaut est <code>JKS</code> .
CCC	Facultatif. Indique si un canal de commande de suppression (non chiffré) est utilisé lorsque l'authentification est terminée. La valeur par défaut est <code>false</code> , ce qui signifie que le canal de commande reste chiffré pendant toute la durée de la session FTPS. Cet attribut est applicable uniquement lorsque <code>ftpsType</code> est défini sur <code>explicit</code> .
protFirst	Facultatif. Indique si les commandes USER/PASS sont émises vers le serveur FTPS avant ou après les commandes PBSZ/PROT . La valeur par défaut est <code>false</code> , ce qui signifie que les commandes USER/PASS sont envoyées en premier, suivies des commandes PBSZ/PROT . Cet attribut est applicable uniquement lorsque <code>ftpsType</code> est défini sur <code>explicit</code> .
auth	Facultatif. Indique le protocole spécifié dans le cadre de la commande AUTH . Un protocole spécifié sera essayé en premier, puis la valeur par défaut est d'essayer TLS, SSL, TLS-Cou TLS-P jusqu'à ce que le serveur FTPS ne soit pas rejeté avec un code de réponse 504. Cet attribut est applicable uniquement lorsque <code>ftpsType</code> est défini sur <code>explicit</code> .

< limites >

Elément de conteneur pour les attributs communs à tous les types de serveur et pour les attributs spécifiques à un type de serveur:

Attributs de limite générale qui s'appliquent à tous les types de serveur de fichiers de protocole:

Attribut	Description
maxListFileNames	Facultatif. Nombre maximal de noms collectés lors de l'analyse d'un répertoire sur le serveur de fichiers de protocole à la recherche de noms de fichier. La valeur par défaut est <code>999999999</code> .
maxListDirectoryLevels	Facultatif. Nombre maximal de niveaux de répertoire sur le serveur de protocole pour l'analyse récursive des noms de fichier. La valeur par défaut est <code>1000</code> .
maxReconnectRelance (Cet attribut est désormais obsolète.)	Obsolète. Cet attribut n'est pas pris en charge sous IBM WebSphere MQ 7.5 ou sous IBM WebSphere MQ File Transfer Edition 7.0.2 ou version ultérieure. Facultatif. Nombre maximal de fois où un serveur de protocole tente de se reconnecter avant que l'agent de pont de protocole n'arrête d'essayer. La valeur par défaut est <code>2</code> .
Période reconnectWait	Obsolète. Cet attribut n'est pas pris en charge sous IBM WebSphere MQ 7.5 ou sous IBM WebSphere MQ File Transfer Edition 7.0.2 ou version ultérieure.

Attribut	Description
(Cet attribut est désormais obsolète.)	Facultatif. Délai d'attente, en secondes, avant la tentative de reconnexion. La valeur par défaut est de 10 secondes.
maxSessions	Facultatif. Nombre maximal de sessions pour le serveur de protocole. Ce nombre doit être supérieur ou égal à la somme du nombre maximal de transferts source et de destination pour l'agent de pont de protocole. La valeur par défaut correspond à la somme des valeurs des propriétés d'agent maxSourceTransferts, maxDestinationTransferts et maxCommandHandlerThreads, plus 1. Si ces trois propriétés utilisent leurs valeurs par défaut de 25, 25 et 5, la valeur par défaut de maxSessions est 56.
socketTimeout	Facultatif. Délai d'attente du socket en secondes. La valeur de cet attribut est utilisée lors de la diffusion en flux de fichiers. La valeur par défaut est fixée à 30 secondes.

Attribut de limite qui s'applique uniquement aux serveurs SFTP:

Attribut	Description
connectionTimeout	Facultatif. Délai d'attente, en secondes, d'une réponse du serveur de fichiers de protocole à une demande de connexion. Un délai d'expiration indique que le serveur de fichiers de protocole n'est pas disponible. La valeur par défaut est de 30 secondes.
cipherList	<p>Facultatif. Indique une liste de chiffrements séparés par des virgules qui sont utilisés pour communiquer entre l'agent de pont de protocole et le serveur SFTP. Les chiffrements sont appelés dans l'ordre dans lequel ils sont spécifiés dans cette liste. Le chiffrement doit être disponible sur le serveur et sur le client avant de pouvoir être utilisé.</p> <p>Les chiffrements pris en charge par l'agent de pont de protocole sont les suivants:</p> <ul style="list-style-type: none"> • fellation-cbc • 3des-cbc • aes128-cbc • aes192-cbc • aes256-cbc • aes128-ctr • aes192-ctr • aes256-ctr • 3des-ctr • arc quatre • arcfour128 • arcfour256 <p>Par défaut, la liste des chiffrements utilisés par les agents de pont de protocole est aes128-cbc , aes192-cbc , aes256-cbc.</p>

Format de fichier de données d'identification Connect:Direct

Le fichier ConnectDirectCredentials.xml du répertoire de configuration Managed File Transfer Agent définit les noms d'utilisateur et les données d'identification utilisés par l'agent Connect:Direct pour s'autoriser avec un noeud Connect:Direct .

Le fichier `ConnectDirectCredentials.xml` doit être conforme au schéma `ConnectDirectCredentials.xsd`. Le document de schéma `ConnectDirectCredentials.xsd` se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/schema` de l'installation MQMFT. Un exemple de fichier `ConnectDirectCredentials.xml` se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/credentials` de l'installation MQMFT.

Le fichier `ConnectDirectCredentials.xml` est rechargé périodiquement par l'agent et toute modification valide apportée au fichier aura une incidence sur le comportement de l'agent. L'intervalle de rechargement par défaut est de 30 secondes. Cet intervalle peut être modifié en spécifiant la propriété d'agent `xmlConfigReloadInterval` dans le fichier `agent.properties`.

Schéma

Le schéma suivant décrit les éléments valides dans le fichier `ConnectDirectCredentials.xml`.

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  This schema defines the format of the XML file that is located in the agent properties
  directory of a Connect:Direct bridge agent. The XML file ConnectDirectCredentials.xml
  is used by the default credential validation of the Connect:Direct bridge.
  For more information, see the WebSphere MQ InfoCenter
-->

<schema targetNamespace="http://wmqfte.ibm.com/ConnectDirectCredentials"
  elementFormDefault="qualified"
  xmlns="https://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://wmqfte.ibm.com/ConnectDirectCredentials"

  <!--
    <?xml version="1.0" encoding="UTF-8"?>

    <tns:credentials xmlns:tns="http://wmqfte.ibm.com/ConnectDirectCredentials"
      xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://wmqfte.ibm.com/ConnectDirectCredentials
        ConnectDirectCredentials.xsd">
      <tns:agent name="CDAGENT01">
        <tns:pnode name="cdnode*" pattern="wildcard">
          <tns:user name="MUSR_.*"
            ignorecase="true"
            pattern="regex"
            cdUserId="bob"
            cdPassword="passw0rd"
            pnodeUserId="bill"
            pnodePassword="alacazam">
          <tns:snode name="cdnode2" pattern="wildcard" userId="sue" password="foo"/>
          </tns:user>
        </tns:pnode>
      </tns:agent>
    </tns:credentials>

    -->

    <element name="credentials" type="tns:credentialsType"/>

    <complexType name="credentialsType">
      <sequence>
        <element name="agent" type="tns:agentType" minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
    </complexType>

    <complexType name="agentType">
      <sequence>
        <element name="pnode" type="tns:pnodeType" minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="name" type="string" use="required"/>
    </complexType>

    <complexType name="pnodeType">
      <sequence>
        <element name="user" type="tns:userType" minOccurs="0" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="name" type="string" use="required"/>
      <attribute name="pattern" type="tns:patternType" use="optional"/>
  -->

```

```

</complexType>

<complexType name="userType">
  <sequence>
    <element name="snode" type="tns:snodeType" minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
  <attribute name="name" type="string" use="required"/>
  <attribute name="ignorecase" type="boolean" use="optional"/>
  <attribute name="pattern" type="tns:patternType" use="optional"/>
  <attribute name="cdUserId" type="string" use="optional"/>
  <attribute name="cdUserIdCipher" type="string" use="optional"/>
  <attribute name="cdPassword" type="string" use="optional"/>
  <attribute name="cdPasswordCipher" type="string" use="optional"/>
  <attribute name="pnodeUserId" type="string" use="optional"/>
  <attribute name="pnodeUserIdCipher" type="string" use="optional"/>
  <attribute name="pnodePassword" type="string" use="optional"/>
  <attribute name="pnodePasswordCipher" type="string" use="optional"/>
</complexType>

<complexType name="snodeType">
  <attribute name="name" type="string" use="required"/>
  <attribute name="pattern" type="tns:patternType" use="optional"/>
  <attribute name="userId" type="string" use="optional"/>
  <attribute name="userIdCipher" type="string" use="optional"/>
  <attribute name="password" type="string" use="optional"/>
  <attribute name="passwordCipher" type="string" use="optional"/>
</complexType>

<simpleType name="patternType">
  <restriction base="string">
    <enumeration value="regex"/>
    <enumeration value="wildcard"/>
  </restriction>
</simpleType>
</schema>

```

Description du fichier ConnectDirectCredentials.xml

Les éléments et attributs utilisés dans le fichier ConnectDirectCredentials.xml sont décrits dans la liste suivante.

< données_d'identification >

Élément de groupe contenant des éléments qui décrivent les données d'identification utilisées par un agent de pont Connect:Direct pour se connecter à un noeud Connect:Direct .

< agent >

Élément de groupe contenant des éléments pour les définitions < pnode > d'un agent nommé.

< noeudp >

Noeud principal (PNODE) dans le transfert Connect:Direct . Ce noeud lance la connexion au noeud secondaire (SNODE).

Attribut	Description
nom	Nom du noeud Connect:Direct . La valeur de cet attribut peut être un modèle qui correspond à de nombreux noms de noeud.
schéma	Indique le type de modèle utilisé pour la valeur de l'attribut de nom. Les valeurs admises pour l'attribut pattern sont les suivantes: <ul style="list-style-type: none"> • joker-les caractères génériques sont utilisés • regex-des expressions régulières Java sont utilisées

< utilisateur >

Utilisateur IBM MQ qui soumet la demande de transfert.

Attribut	Description
nom	Nom d'utilisateur utilisé avec Managed File Transfer. La valeur de cet attribut peut être un modèle qui correspond à de nombreux noms d'utilisateur.

Attribut	Description
ignorecase	Indique si la casse du nom est ignorée. Les valeurs admises pour l'attribut ignorecase sont les suivantes: <ul style="list-style-type: none"> • true-le nom n'est pas sensible à la casse • false-le nom est sensible à la casse
schéma	Indique le type de modèle utilisé pour la valeur de l'attribut de nom. Les valeurs admises pour l'attribut pattern sont les suivantes: <ul style="list-style-type: none"> • joker-les caractères génériques sont utilisés • regex-des expressions régulières Java sont utilisées
ID cdUserou cdUserIdCipher	Nom d'utilisateur utilisé par le pont Connect:Direct pour se connecter à son noeud Connect:Direct associé. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
Chiffrement cdPassword ou cdPassword	Mot de passe associé au nom d'utilisateur spécifié par l'attribut d'ID cdUser. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
pnodeUserID ou pnodeUserIdCipher	Nom d'utilisateur utilisé par le noeud principal Connect:Direct . Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
Chiffrement pnodePassword ou pnodePassword	Mot de passe associé au nom d'utilisateur spécifié par l'attribut d'ID pnodeUser. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.

< noeuds >

Le noeud Connect:Direct qui joue le rôle de noeud secondaire (SNOE) lors du transfert de fichiers Connect:Direct .

Attribut	Description
nom	Nom du noeud Connect:Direct . La valeur de cet attribut peut être un modèle qui correspond à de nombreux noms de noeud.
schéma	Indique le type de modèle utilisé pour la valeur de l'attribut de nom. Les valeurs admises pour l'attribut pattern sont les suivantes: <ul style="list-style-type: none"> • joker-les caractères génériques sont utilisés • regex-des expressions régulières Java sont utilisées
Chiffrement userId ou userId	Nom d'utilisateur utilisé pour se connecter à ce noeud lors d'un transfert de fichier. Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.
mot de passe ou passwordCipher	Mot de passe associé au nom d'utilisateur spécifié par l'attribut userId . Si la commande fteObfuscate a été utilisée, la version de chiffrement de l'attribut doit être utilisée.

Exemple

Dans cet exemple, l'agent de pont Connect:Direct se connecte au Connect:Direct noeud pnode1. Lorsqu'un utilisateur IBM MQ dont le nom d'utilisateur commence par le préfixe fteuser suivi d'un seul caractère, par exemple fteuser12, demande un transfert impliquant le pont Connect:Direct , l'agent de pont Connect:Direct utilise le nom d'utilisateur cduser et le mot de passe passw0rd pour se connecter au Connect:Direct noeud pnode1. Lorsque le Connect:Direct noeud pnode1 effectue sa partie du transfert, il utilise le nom d'utilisateur pnodeuser et le mot de passe passw0rd1.

Si le nom du noeud secondaire dans le transfert Connect:Direct commence par le préfixe FISH, le noeud pnode1 utilise le nom d'utilisateur fishuser et le mot de passe passwd2 pour se connecter au noeud secondaire. Si le nom du noeud secondaire dans le transfert Connect:Direct commence par le préfixe CHIPS, le noeud pnode1 utilise le nom d'utilisateur chipsuser et le mot de passe passwd3 pour se connecter au noeud secondaire.

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:credentials xmlns:tns="http://wmqfte.ibm.com/ConnectDirectCredentials"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ConnectDirectCredentials
ConnectDirectCredentials.xsd">
  <tns:agent name="CDAGENT01">
    <tns:pnode name="pnode1" pattern="wildcard">
      <tns:user name="fteuser?" pattern="wildcard" ignorecase="true"
        cdUserId="cduser" cdPassword="passwd"
        pnodeUserId="pnodeuser" pnodePassword="passwd1">
      <tns:snode name="FISH*" pattern="wildcard"
        userId="fishuser" password="passwd2"/>
      <tns:snode name="CHIPS*" pattern="wildcard"
        userId="chipsuser" password="passwd3"/>
    </tns:user>
  </tns:pnode>
</tns:agent>
</tns:credentials>
```

Format de fichier des définitions de processus Connect:Direct

Le fichier ConnectDirectProcessDefinitions.xml du répertoire de configuration de l'agent de pont Connect:Direct spécifie le processus Connect:Direct défini par l'utilisateur à démarrer dans le cadre du transfert de fichiers.

Le fichier ConnectDirectProcessDefinitions.xml doit être conforme au schéma ConnectDirectProcessDefinitions.xsd. Le document de schéma ConnectDirectProcessDefinitions.xsd se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/schema` de l'installation MFT. Un fichier modèle ConnectDirectProcessDefinitions.xml est créé par la commande **fteCreateCDAgent** dans le répertoire de configuration de l'agent.

Le fichier ConnectDirectProcessDefinitions.xml est rechargé périodiquement par l'agent et toute modification valide apportée au fichier aura une incidence sur le comportement de l'agent. L'intervalle de rechargement par défaut est de 30 secondes. Cet intervalle peut être modifié en spécifiant la propriété d'agent `xmlConfigReloadInterval` dans le fichier `agent.properties`.

Schéma

Le schéma suivant décrit les éléments valides dans le fichier ConnectDirectProcessDefinitions.xml.

```
<schema targetNamespace="http://wmqfte.ibm.com/ConnectDirectProcessDefinitions"
  elementFormDefault="qualified"
  xmlns="https://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://wmqfte.ibm.com/ConnectDirectProcessDefinitions">
  <element name="cdprocess" type="tns:cdprocessType"></element>
  <complexType name="cdprocessType">
    <sequence>
      <element name="processSet" type="tns:processSetType"
        minOccurs="0" maxOccurs="unbounded"></element>
    </sequence>
  </complexType>
  <complexType name="processSetType">
    <sequence>
      <element name="condition" type="tns:conditionType"
        minOccurs="0" maxOccurs="1"/>
      <element name="process" type="tns:processType"
        minOccurs="1" maxOccurs="1"/>
    </sequence>
```

```

</complexType>

<complexType name="conditionType">
  <choice minOccurs="0" maxOccurs="unbounded">
    <element name="match" type="tns:matchType"/>
    <element name="defined" type="tns:definedType"/>
  </choice>
</complexType>

<complexType name="matchType">
  <attribute name="variable" type="string" use="required"/>
  <attribute name="value" type="string" use="required"/>
  <attribute name="pattern" type="tns:patternType" use="optional"/>
</complexType>

<complexType name="definedType">
  <attribute name="variable" type="string" use="required"/>
</complexType>

<complexType name="processType">
  <sequence>
    <element name="preTransfer" type="tns:transferType"
      minOccurs="0" maxOccurs="1"/>
    <element name="transfer" type="tns:transferType"
      minOccurs="0" maxOccurs="1"/>
    <element name="postTransferSuccess" type="tns:transferType"
      minOccurs="0" maxOccurs="1"/>
    <element name="postTransferFailure" type="tns:transferType"
      minOccurs="0" maxOccurs="1"/>
  </sequence>
</complexType>

<complexType name="transferType">
  <attribute name="process" type="string" use="required"/>
</complexType>

<simpleType name="patternType">
  <restriction base="string">
    <enumeration value="regex"/>
    <enumeration value="wildcard"/>
  </restriction>
</simpleType>

</schema>

```

Description du fichier ConnectDirectProcessDefinitions.xml

Les éléments et attributs utilisés dans le fichier ConnectDirectProcessDefinitions.xml sont décrits dans la liste suivante.

cdProcess

Élément racine du document XML.

processSet

Élément de groupe contenant toutes les informations sur un ensemble de processus définis par l'utilisateur.

condition

Élément de groupe contenant les conditions par rapport auxquelles un transfert est testé pour déterminer si l'ensemble de processus contenu dans l'élément processSet est utilisé.

correspond à

Condition qui teste si la valeur d'une variable correspond à une valeur donnée.

Attribut	Description
variable	Indique une variable. La valeur de cette variable est comparée à la valeur de l'attribut value . La variable est un symbole intrinsèque. Pour plus d'informations, voir la section «Variables de substitution à utiliser avec les processus Connect:Direct définis par l'utilisateur», à la page 680.
valeur	Indique un modèle à mettre en correspondance avec la valeur de la variable spécifiée par l'attribut variable .

Attribut	Description
schéma	Indique le type de modèle utilisé pour la valeur de l'attribut value . Les valeurs admises pour l'attribut pattern sont les suivantes: <ul style="list-style-type: none"> • wildcard -les caractères génériques sont utilisés • regex -Les expressions régulières Java sont utilisées Cet attribut est facultatif et la valeur par défaut est wildcard.

défini

Condition qui teste si une variable a été définie.

Attribut	Description
variable	Indique une variable. Si cette variable existe, la condition de correspondance est satisfaite. La variable est un symbole intrinsèque. Pour plus d'informations, voir la section «Variables de substitution à utiliser avec les processus Connect:Direct définis par l'utilisateur», à la page 680.

processus

Élément de groupe contenant les informations sur l'emplacement des processus Connect:Direct à appeler lorsqu'une correspondance est trouvée.

transfert

Processus Connect:Direct à appeler lors d'une demande de transfert.

Attribut	Description
processus	Facultatif. Indique le nom d'un fichier contenant un processus Connect:Direct à appeler lors d'une demande de transfert. Le chemin d'accès au fichier est relatif au répertoire de configuration de l'agent de pont Connect:Direct . Cet attribut est facultatif. Par défaut, un processus généré par MFTTest utilisé. Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.

Exemple

Dans cet exemple, il existe trois éléments processSet .

Le premier élément processSet indique que si une demande de transfert comporte une variable **%FTESNODE** dont la valeur correspond au modèle Client* et une variable **%FTESUSER** dont la valeur est Admin, l'agent de pont Connect:Direct soumet le processus Connect:Direct situé dans le *agent_configuration_directory/AdminClient.cdp* dans le cadre du transfert.

Le second élément processSet indique que si une demande de transfert comporte une variable **%FTESNODE** avec une valeur qui correspond au modèle Client*, l'agent de pont Connect:Direct soumet le processus Connect:Direct situé dans *agent_configuration_directory/Client.cdp* dans le cadre du transfert. L'agent de pont Connect:Direct lit les éléments processSet dans l'ordre dans lequel ils sont définis, et s'il trouve une correspondance, il utilise la première correspondance et ne recherche pas une autre correspondance. Pour les demandes de transfert qui correspondent aux conditions du premier et du second processSet, l'agent de pont Connect:Direct appelle uniquement les processus spécifiés par le premier processSet.

Le troisième élément processSet n'a pas de conditions et correspond à tous les transferts. Si la demande de transfert ne correspond pas aux conditions de la première ou de la deuxième processSet, l'agent de pont Connect:Direct soumet le processus Connect:Direct spécifié par la troisième condition. Ce processus se trouve dans le fichier *agent_configuration_directory/Default.cdp* dans le cadre du transfert.

```

<?xml version="1.0" encoding="UTF-8"?>
<tns:cdprocess xmlns:tns="http://wmqfte.ibm.com/ConnectDirectProcessDefinitions"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ConnectDirectProcessDefinitions
ConnectDirectProcessDefinitions.xsd">

  <tns:processSet>
    <tns:condition>
      <tns:match variable="%FTESNODE" value="Client*" pattern="wildcard"/>
      <tns:match variable="%FTESUSER" value="Admin" pattern="wildcard"/>
    </tns:condition>
    <tns:process>
      <tns:transfer process="AdminClient.cdp"/>
    </tns:process>
  </tns:processSet>

  <tns:processSet>
    <tns:condition>
      <tns:match variable="%FTESNODE" value="Client*" pattern="wildcard"/>
    </tns:condition>
    <tns:process>
      <tns:transfer process="Client.cdp"/>
    </tns:process>
  </tns:processSet>

  <tns:processSet>
    <tns:process>
      <tns:transfer process="Default.cdp"/>
    </tns:process>
  </tns:processSet>
</tns:cdprocess>

```

Format du fichier de propriétés du noeud Connect:Direct

Le fichier `ConnectDirectNodeProperties.xml` du répertoire de configuration de l'agent de pont `Connect:Direct` spécifie des informations sur les noeuds `Connect:Direct` distants impliqués dans un transfert de fichiers.

Le fichier `ConnectDirectNodeProperties.xml` doit être conforme au schéma `ConnectDirectNodeProperties.xsd`. Le document de schéma `ConnectDirectNodeProperties.xsd` se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/schema` de l'installation MFT. Un fichier modèle `ConnectDirectNodeProperties.xml` est créé par la commande **fteCreateCDAgent** dans le répertoire de configuration de l'agent.

Le fichier `ConnectDirectNodeProperties.xml` est rechargé périodiquement par l'agent et toute modification valide apportée au fichier aura une incidence sur le comportement de l'agent. L'intervalle de rechargement par défaut est de 30 secondes. Cet intervalle peut être modifié en spécifiant la propriété d'agent `xmlConfigReloadInterval` dans le fichier `agent.properties`.

Schéma

Le schéma suivant décrit les éléments valides dans le fichier `ConnectDirectNodeProperties.xml`.

```

<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="http://wmqfte.ibm.com/ConnectDirectNodeProperties"
  elementFormDefault="qualified"
  xmlns="https://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://wmqfte.ibm.com/ConnectDirectNodeProperties">

  <element name="nodeProperties" type="tns:nodePropertiesType"></element>

  <complexType name="nodePropertiesType">
    <sequence>
      <element name="credentialsFile" type="tns:credentialsFileName" minOccurs="0" maxOccurs="1"/>
      <element name="node" type="tns:nodeType" minOccurs="0" maxOccurs="unbounded"></element>
    </sequence>
  </complexType>

  <complexType name="nodeType">
    <attribute name="name" type="string" use="required"/>
    <attribute name="pattern" type="tns:patternType" use="optional"/>
    <attribute name="type" type="string" use="required"/>
  </complexType>

```

```

</complexType>

<simpleType name="patternType">
  <restriction base="string">
    <enumeration value="regex"/>
    <enumeration value="wildcard"/>
  </restriction>
</simpleType>

</schema>

```

Description du fichier ConnectDirectNodeProperties.xml

Les éléments et attributs utilisés dans le fichier ConnectDirectNodeProperties.xml sont décrits dans la liste suivante.

nodeProperties

Élément racine du document XML.

credentialsFile

Chemin d'accès au fichier de données d'identification dans lequel les informations sensibles sont stockées. Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir [«Utilisation des variables d'environnement dans les propriétés MFT»](#), à la page 547

noeud

Indique un ou plusieurs noeuds Connect:Direct .

Attribut	Description
nom	Modèle qui identifie les noms des noeuds Connect:Direct qui utilisent les définitions spécifiées par l'élément node. La correspondance de modèle n'est pas sensible à la casse.
schéma	Indique le type de modèle utilisé pour la valeur de l'attribut name . Les valeurs admises pour l'attribut pattern sont les suivantes: <ul style="list-style-type: none"> • joker-les caractères génériques sont utilisés • regex-des expressions régulières Java sont utilisées Pour plus d'informations sur les types d'expression régulière utilisés par MFT, voir «Expressions régulières utilisées par MFT» , à la page 679.
type	Indique le type de système d'exploitation du ou des noeuds Connect:Direct qui correspondent au modèle indiqué par l'attribut name . Les valeurs admises pour l'attribut type sont les suivantes: <ul style="list-style-type: none"> • Windows -le noeud s'exécute sur Windows • UNIX -le noeud s'exécute sous UNIX ou Linux •  z/OS, zos, os/390ou os390 -le noeud s'exécute sur z/OS La valeur de cet attribut n'est pas sensible à la casse.

Exemple

Dans cet exemple, le nom de fichier des données d'identification Connect:Direct est spécifié sous la forme ConnectDirectCredentials.xml. L'exemple de code spécifie les connexions de plateforme suivantes:

- Tous les noeuds Connect:Direct dont le nom commence par "cdnodew" s'exécutent sur la plateforme Windows .
- Tous les noeuds Connect:Direct dont le nom commence par "cdnodeu" s'exécutent sur la plateforme UNIX .
- Tous les noeuds Connect:Direct dont le nom commence par "cdnodez" s'exécutent sur la plateforme z/OS .

- Tous les autres noeuds Connect:Direct s'exécutent sur la plateforme UNIX .

L'agent de pont Connect:Direct recherche les correspondances entre le début et la fin du fichier et utilise la première correspondance qu'il trouve.

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:nodeProperties xmlns:tns="http://wmqfte.ibm.com/ConnectDirectNodeProperties"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ConnectDirectNodeProperties
    ConnectDirectNodeProperties.xsd">

  <tns:credentialsFile path="ConnectDirectCredentials.xml"/>
  <tns:node name="cdnodew*" pattern="wildcard" type="windows"/>
  <tns:node name="cdnodeu.*" pattern="regex" type="unix"/>
  <tns:node name="cdnodez*" pattern="wildcard" type="zos"
  <tns:node name="*" pattern="wildcard" type="unix"/>

</tns:nodeProperties>
```

Fichier de schéma fteutils.xsd

Ce schéma définit les éléments et les types utilisés par de nombreux autres schémas Managed File Transfer .

Schéma

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
@start_non_restricted_prolog@
Version: %Z% %I% %W% %E% %U% [%H% %T%]

Licensed Materials - Property of IBM

5724-H72

Copyright IBM Corp. 2008, 2023. All Rights Reserved.

US Government Users Restricted Rights - Use, duplication or
disclosure restricted by GSA ADP Schedule Contract with
IBM Corp.
@end_non_restricted_prolog@
-->

<!--
This schema defines elements and types used by many of the other MQMFT schemas.
For more information about MQMFT XML message formats, see
https://www.ibm.com/docs/SSEP7X_7.0.4/com.ibm.wmqfte.doc/message_formats.htm
-->
<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema">
  <!--
    Defines the version type 1.00 - 99.00
  -->
  <transaction version= 1.00
  -->
  <xsd:simpleType name="versionType">
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="[0-9]+\.[0-9][0-9]"/>
    </xsd:restriction>
  </xsd:simpleType>
  <!--
    Defines the transaction reference
  -->
  <transaction version= 1.00 ID="414d5120514d312020202020202020205ecf0a4920011802"
  -->
  <xsd:simpleType name="IDType">
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="[0-9a-fA-F]{48}"/>
    </xsd:restriction>
  </xsd:simpleType>
  <!--
    This is an alias for hostUserIDType.
    Here to allow addition of attributes on originator elements
  -->
  <xsd:complexType name="origRequestType">
    <xsd:complexContent>
      <xsd:extension base="hostUserIDType">
        <xsd:sequence>
          <xsd:element name="webBrowser" type="xsd:string" minOccurs="0"
-->
```

```

maxOccurs="1"/>
    <xsd:element name="webUserID" type="xsd:string" minOccurs="0"
maxOccurs="1"/>
    </xsd:sequence>
  </xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<!--
  Defines a Delete originator as a machine and user pair
  <hostName>myMachine</hostName>
  <userName>myUserId</userName>
-->
<xsd:complexType name="origDeleteType">
  <xsd:sequence>
    <xsd:element name="delete" type="hostUserIDType" maxOccurs="1" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<!--
  Defines a machine, user, MQMD userID triple
  <hostName>myMachine</hostName>
  <userID>myUserId</userID>
  <mqmdUserID>MQMDUSERID</mqmdUserID>
-->
<xsd:complexType name="hostUserIDType">
  <xsd:sequence>
    <xsd:element name="hostName" type="xsd:string" minOccurs="0" maxOccurs="1"/>
    <xsd:element name="userID" type="xsd:string" minOccurs="0" maxOccurs="1"/>
    <xsd:element name="mqmdUserID" type="xsd:string" minOccurs="0" maxOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>
<!--
  Define the destinationAgent with agent and queue manager name
  <destinationAgent agent="there" QMgr="far" agentType="BRIDGE" bridgeURL="ftp://
server.example.net:21" bridgeNode="DEST_NODE"/>
  optional agentType attribute expected to be one of STANDARD, BRIDGE, WEB_GATEWAY,
EMBEDDED, CD_BRIDGE
-->
<xsd:complexType name="agentType">
  <xsd:attribute name="agent" type="xsd:string" use="required"/>
  <xsd:attribute name="agentType" type="xsd:string" use="optional"/>
  <xsd:attribute name="QMgr" type="xsd:string" use="optional"/>
  <xsd:attribute name="bridgeURL" type="xsd:string" use="optional"/>
  <xsd:attribute name="bridgeNode" type="xsd:string" use="optional"/>
  <xsd:attribute name="pnode" type="xsd:string" use="optional"/>
  <xsd:attribute name="snode" type="xsd:string" use="optional"/>
</xsd:complexType>
<!--
  Defines the status type; attr/resultCode and 0 or many supplements
  There may also be additional command specific data, either: transfer, ping or call data
  <status resultCode="8011">
    <supplement>Azionamento del USB</supplement>
    <supplement>morto come norweign azzurro</supplement>
  </status>
-->
<xsd:complexType name="statusType">
  <xsd:sequence>
    <xsd:element name="supplement" type="xsd:string" maxOccurs="unbounded"
minOccurs="0"/>
    <xsd:choice>
      <xsd:element name="fileSpace" type="fileSpaceReplyType" minOccurs="0"
maxOccurs="1"/>
    </xsd:choice>
  </xsd:sequence>
  <xsd:attribute name="resultCode" type="resultCodeType" use="required"/>
</xsd:complexType>
<!--
  Defines the fileSpace type for use with communication between a web agent
and a web gateway
  <fileSpace name="" location=""><Quota bytes=""></fileSpace>
-->
<xsd:complexType name="fileSpaceReplyType">
  <xsd:attribute name="name" use="required" type="xsd:string"/>
  <xsd:attribute name="location" use="required" type="xsd:string"/>
  <xsd:attribute name="quota" use="required" type="xsd:long"/>
</xsd:complexType>
<!--
  Defines the destinationAgent with agent and queue manager name, plus connection
details.
  <destinationAgent agent="there" QMgr="far"/>
-->
<xsd:complexType name="agentClientType">
  <xsd:attribute name="agent" type="xsd:string" use="required"/>

```

```

    <xsd:attribute name="QMgr" type="xsd:string" use="optional"/>
    <xsd:attribute name="hostName" type="xsd:string" use="optional"/>
    <xsd:attribute name="portNumber" type="xsd:nonNegativeInteger" use="optional"/>
    <xsd:attribute name="channel" type="xsd:string" use="optional"/>
</xsd:complexType>
<!--
    Defines the fileURI type as string
    <file encoding="UTF8" EOL="CR">C:/from/here.txt</file>
-->
<xsd:complexType name="fileType">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute name="filesystem" type="filesystemNameType" use="optional"/>
      <xsd:attribute name="alias" type="xsd:string" use="optional"/>
      <xsd:attribute name="encoding" type="encodingType" use="optional"/>
      <xsd:attribute name="EOL" type="EOLType" use="optional"/>
      <xsd:attribute name="size" type="xsd:long" use="optional"/>
      <xsd:attribute name="last-modified" type="xsd:dateTime" use="optional"/>
      <xsd:attribute name="delimiter" type="xsd:string" use="optional"/>
      <xsd:attribute name="delimiterType" type="xsd:string" use="optional"/>
      <xsd:attribute name="delimiterPosition" type="delimiterPositionType"
use="optional"/>
      <xsd:attribute name="includeDelimiterInFile" type="xsd:boolean" use="optional"/>
      <xsd:attribute name="keepTrailingSpaces" type="xsd:boolean" use="optional"/>
      <xsd:attribute name="truncateRecords" type="xsd:boolean" use="optional"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
<!--
    Defines the filesystem type as string
    <filesystem>
      <name>tarquin</name>
    </filesystem>
-->
<xsd:complexType name="filesystemType">
  <xsd:sequence>
    <xsd:element name="name" type="filesystemNameType"/>
  </xsd:sequence>
</xsd:complexType>
<!--
    Defines a name element
    <name>bob</name>
-->
<xsd:simpleType name="filesystemNameType">
  <xsd:restriction base="xsd:string"/>
</xsd:simpleType>
<!--
    Defines the accepted choices for the persistent attribute.
-->
<xsd:simpleType name="persistenceType">
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="true"/>
    <xsd:enumeration value="false"/>
    <xsd:enumeration value="qdef"/>
  </xsd:restriction>
</xsd:simpleType>
<!--
    Defines the queueURI type as string with all supported attributes.
    <queue>QUEUE@QM</queue>
-->
<xsd:complexType name="queueType">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute name="persistent" type="persistenceType" use="optional"/>
      <xsd:attribute name="eofMarker" type="xsd:boolean" use="optional"/>
      <xsd:attribute name="setMqProps" type="xsd:boolean" use="optional"/>
      <xsd:attribute name="split" type="xsd:boolean" use="optional"/>
      <xsd:attribute name="useGroups" type="xsd:boolean" use="optional"/>
      <xsd:attribute name="delimiter" type="xsd:string" use="optional"/>
      <xsd:attribute name="delimiterType" type="xsd:string" use="optional"/>
      <xsd:attribute name="delimiterPosition" type="delimiterPositionType"
use="optional"/>
      <xsd:attribute name="includeDelimiterInMessage" type="xsd:boolean"
use="optional"/>
      <xsd:attribute name="groupId" type="groupIdType" use="optional"/>
      <xsd:attribute name="messageId" type="messageIdType" use="optional"/>
      <xsd:attribute name="messageCount" type="xsd:nonNegativeInteger"
use="optional"/>
      <xsd:attribute name="messageLength" type="xsd:nonNegativeInteger"
use="optional"/>
      <xsd:attribute name="waitTime" type="xsd:nonNegativeInteger" use="optional"/>
      <xsd:attribute name="encoding" type="encodingType" use="optional"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>

```

```

        <xsd:attribute name="EOL" type="EOLType" use="optional"/>
        <xsd:attribute name="unrecognisedCodePage" type="unrecognisedCodePageType"
use="optional"/>
    </xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
<!--
    Defines the accepted values for the delimiterPosition attribute.
-->
<xsd:simpleType name="delimiterPositionType">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="postfix"/>
        <xsd:enumeration value="prefix"/>
    </xsd:restriction>
</xsd:simpleType>
<!--
    Defines the groupId type
    <queue groupId="414d5120514d31202020202020202020205ecf0a4920011802">
    Also allow a substitution variable of the form ${variable}
-->
<xsd:simpleType name="groupIdType">
    <xsd:restriction base="xsd:string">
        <xsd:pattern value="[0-9a-fA-F]{48}|\${.*}" />
    </xsd:restriction>
</xsd:simpleType>
<!--
    Defines the messageId type
    <queue messageId="414d5120514d31202020202020202020205ecf0a4920011802">
    Also allow a substitution variable of the form ${variable}
-->
<xsd:simpleType name="messageIdType">
    <xsd:restriction base="xsd:string">
        <xsd:pattern value="[0-9a-fA-F]{48}|\${.*}" />
    </xsd:restriction>
</xsd:simpleType>
<!-- Defines the accepted values for the unrecognisedCodePage attribute. -->
<xsd:simpleType name="unrecognisedCodePageType">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="fail"/>
        <xsd:enumeration value="binary"/>
    </xsd:restriction>
</xsd:simpleType>
<!--
    Defines a single source file reference
    <source type="file" recursive="false" disposition="leave">
        <file>filename</file>
    </source>
-->
<xsd:complexType name="fileSourceType">
    <xsd:sequence>
        <xsd:choice>
            <xsd:element name="file" type="fileType"/>
            <xsd:element name="queue" type="queueType"/>
        </xsd:choice>
    </xsd:sequence>
    <xsd:attribute name="type" type="SourceType" use="optional"/>
    <xsd:attribute name="recursive" type="xsd:boolean" use="optional"/>
    <xsd:attribute name="disposition" type="sourceDispositionType" use="optional"/>
    <xsd:attribute name="correlationString1" type="xsd:string" use="optional"/>
    <xsd:attribute name="correlationNum1" type="xsd:nonNegativeInteger" use="optional"/>
    <xsd:attribute name="correlationBoolean1" type="xsd:boolean" use="optional"/>
</xsd:complexType>
<!--
    Defines the enumeration values for source type
    type="file|queue"
-->
<xsd:simpleType name="SourceType">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="file"/>
        <xsd:enumeration value="directory"/>
        <xsd:enumeration value="queue"/>
        <xsd:enumeration value="dataset"/>
        <xsd:enumeration value="pds"/>
        <xsd:enumeration value="filesystem"/>
    </xsd:restriction>
</xsd:simpleType>
<!--
    Defines the enumeration values for source disposition
    disposition="leave|delete"
-->
<xsd:simpleType name="sourceDispositionType">
    <xsd:restriction base="xsd:token">

```

```

        <xsd:enumeration value="leave"/>
        <xsd:enumeration value="delete"/>
    </xsd:restriction>
</xsd:simpleType>
<!--
    Defines a single destination file reference
    <destination type="file" exist="overwrite">
        <file>filename</file>
    </destination/>
-->
<xsd:complexType name="fileDestinationType">
    <xsd:sequence>
        <xsd:choice>
            <xsd:element name="file" type="fileType"/>
            <xsd:element name="filespace" type="filespaceType"/>
            <xsd:element name="queue" type="queueType"/>
        </xsd:choice>
        <xsd:element name="attributes" type="attributeType" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute name="type" type="DestinationType" use="optional"/>
    <xsd:attribute name="exist" type="existType" use="optional"/>
    <xsd:attribute name="correlationString1" type="xsd:string" use="optional"/>
    <xsd:attribute name="correlationNum1" type="xsd:nonNegativeInteger" use="optional"/>
    <xsd:attribute name="correlationBoolean1" type="xsd:boolean" use="optional"/>
</xsd:complexType>
<!--
    Defines the enumeration values for destination file type
    type="file|directory|queue|dataset|pds|filespace"
    'dataset' and 'pds' only apply to z/OS environments.
-->
<xsd:simpleType name="DestinationType">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="file"/>
        <xsd:enumeration value="directory"/>
        <xsd:enumeration value="queue"/>
        <xsd:enumeration value="dataset"/>
        <xsd:enumeration value="pds"/>
        <xsd:enumeration value="filespace"/>
    </xsd:restriction>
</xsd:simpleType>
<!--
    Defines the enumerations values for file exists on destination behavior
    exist="error|overwrite"
-->
<xsd:simpleType name="existType">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="error"/>
        <xsd:enumeration value="overwrite"/>
    </xsd:restriction>
</xsd:simpleType>
<!--
    Defines one or more file attributes
    <destination encoding=? CFLF=?>
        <file>filename</file>
        <attributes>
            <attribute>DIST(MIRRORED,UPDATE)</attribute>
        </attributes>
    </destination/>
-->
<xsd:complexType name="attributeType">
    <xsd:sequence>
        <xsd:element name="attribute" type="xsd:string" maxOccurs="unbounded"
minOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>

<!--
    Defines a single file reference
    <source encoding=? CFLF=?>
        <file>filename</file>
        <checksum method="MD5">3445678</checksum>
    </source/>
    .. or ..
    <destination encoding=? CFLF=?>
        <file>filename</file>
        <checksum method="MD5">3445678</checksum>
    </destination/>
-->
<xsd:complexType name="fileCheckSumType">
    <xsd:sequence>

```

```

        <xsd:element name="file" type="fileType"/>
        <xsd:element name="checksum" type="checksumType" maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
<!--
    Defines the checksum type and method
    <checksum method="MD5|none">3445678</checksum>
-->
<xsd:complexType name="checksumType">
    <xsd:simpleContent>
        <xsd:extension base="xsd:string">
            <xsd:attribute name="method" type="checkSumMethod" use="required"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>
<!--
    Defines the enumeration values for checksumMethod
    <checksum method="MD5|none">3445678</checksum>
    Note: uppercase is used since MD5 is acronym and normally written uppercase.
-->
<xsd:simpleType name="checkSumMethod">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="none"/>
        <xsd:enumeration value="MD5"/>
    </xsd:restriction>
</xsd:simpleType>
<!--
    Defines the enumeration values for agentRole
    agentRole="sourceAgent|destinationAgent"
-->
<xsd:simpleType name="agentRoleType">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="sourceAgent"/>
        <xsd:enumeration value="destinationAgent"/>
        <xsd:enumeration value="callAgent"/>
    </xsd:restriction>
</xsd:simpleType>
<!--
    Defines the enumeration values for mode.
    text, binary or a substitution variable
    <item mode="binary|text|${variableName}">
-->
<xsd:simpleType name="modeType">
    <xsd:restriction base="xsd:string">
        <xsd:pattern value="binary|text|\${.*}" />
    </xsd:restriction>
</xsd:simpleType>
<!--
    Defines the enumeration values for EOL
    <file EOL="LF|CRLF">
-->
<xsd:simpleType name="EOLType">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="LF"/>
        <xsd:enumeration value="CRLF"/>
    </xsd:restriction>
</xsd:simpleType>
<!--
    Defines the encoding type as a string
-->
<xsd:simpleType name="encodingType">
    <xsd:restriction base="xsd:string"/>
</xsd:simpleType>
<!--
    <schedule>
        <submit timebase="source|"admin">2008-12-07T16:07"</submit>
        <repeat>
            <frequency interval="hours">2</frequency>
            <expireTime>2008-12-0816:07</exipreTime>
        </repeat>
    </schedule>
-->
<xsd:complexType name="scheduleType">
    <xsd:sequence>
        <xsd:element name="submit" type="submitType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="repeat" type="repeatType" maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
<!--
    <submit timebase="source|admin|UTC">2008-12-07T16:07"</submit>
-->
<xsd:complexType name="submitType">

```

```

    <xsd:simpleContent>
      <xsd:extension base="noZoneTimeType">
        <xsd:attribute name="timebase" type="timebaseType" use="required"/>
        <xsd:attribute name="timezone" type="xsd:string" use="required"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
<!--
  <repeat>
    <frequency interval="hours">2</frequency>
    ..optionally..
    <expireTime>2008-12-08T16:07</expireTime>
    ..or..
    <expireCount>2</expireCount>
  </repeat>
-->
<xsd:complexType name="repeatType">
  <xsd:sequence>
    <xsd:element name="frequency" type="freqType" maxOccurs="1" minOccurs="1"/>
    <xsd:choice minOccurs="0">
      <xsd:element name="expireTime" type="noZoneTimeType"/>
      <xsd:element name="expireCount" type="positiveIntegerType"/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
<!--
  <frequency interval="hours">2</frequency>
-->
<xsd:complexType name="freqType">
  <xsd:simpleContent>
    <xsd:extension base="positiveIntegerType">
      <xsd:attribute name="interval" type="intervalType" use="required"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
<!--
  Defines positive integer type
  i.e., 1+
-->
<xsd:simpleType name="positiveIntegerType">
  <xsd:restriction base="xsd:integer">
    <xsd:minInclusive value="1"/>
  </xsd:restriction>
</xsd:simpleType>
<!--
  Defines the interval enumeration values of
  "minutes", "hours", "days", "weeks", "months" or "years"
-->
<xsd:simpleType name="intervalType">
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="minutes"/>
    <xsd:enumeration value="hours"/>
    <xsd:enumeration value="days"/>
    <xsd:enumeration value="weeks"/>
    <xsd:enumeration value="months"/>
    <xsd:enumeration value="years"/>
  </xsd:restriction>
</xsd:simpleType>
<!--
  Defines the interval of either "source", "admin" or "UTC"
  source = use timezone of the source Agent.
  admin = use timezone of the administrator executing the command script.
  UTC = Timezone is UTC.
-->
<xsd:simpleType name="timebaseType">
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="source"/>
    <xsd:enumeration value="admin"/>
    <xsd:enumeration value="UTC"/>
  </xsd:restriction>
</xsd:simpleType>
<!--
  Defines a date and time without a time zone (2008-12-08T16:07)
-->
<xsd:simpleType name="noZoneTimeType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[\n\r\t ]*\d{4}\-(0[1-9]|1[0-2])\-(0[1-9]|[1-2][0-9]|
3[0-1])T([0-1][0-9]|2[0-3]):[0-5][0-9]([\+\-]\d{4}|Z)?[\n\r\t ]*"/>
  </xsd:restriction>
</xsd:simpleType>
<!--
  Defines the ID element, e.g. 56

```

```

-->
<xsd:simpleType name="idType">
  <xsd:restriction base="xsd:string"/>
</xsd:simpleType>
<!--
  Defines the resultCode type -2 - 9999
  <status resultCode="8011">
-->
<xsd:simpleType name="resultCodeType">
  <xsd:restriction base="xsd:int">
    <xsd:minInclusive value="-2"/>
    <xsd:maxInclusive value="9999"/>
  </xsd:restriction>
</xsd:simpleType>
<!--
  Define the metaDataSet type comprising one or more key value pairs
  <metaDataSet>
    <metaData key="name">value</metaData>
    <metaData key="name">value</metaData>
  </metaDataSet>
-->
<xsd:complexType name="metaDataSetType">
  <xsd:sequence>
    <xsd:element name="metaData" type="metaDataType" maxOccurs="unbounded"
minOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>
<!--
  Define the metaData type which is made up of a key and a value
  <metaData key="name">value</metaData>
-->
<xsd:complexType name="metaDataType">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute name="key" type="xsd:string" use="required"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
<!--
  Defines containing element for triggers
  <trigger log="yes">
    <fileExist comparison="=" value="Exist">file1</fileExist>
    <fileSize comparison=">=" value="1GB">file1</fileSize>
  </trigger>
-->
<xsd:complexType name="triggerType">
  <xsd:choice minOccurs="0" maxOccurs="unbounded">
    <xsd:element name="fileExist" type="fileExistTriggerType" maxOccurs="unbounded"
minOccurs="1"/>
    <xsd:element name="fileSize" type="fileSizeTriggerType" maxOccurs="unbounded"
minOccurs="1"/>
  </xsd:choice>
  <xsd:attribute name="log" type="logEnabledType" use="required"/>
</xsd:complexType>
<!--
  Defines the file exists trigger type
  <fileExist comparison="=" value="Exist">file1</trigger>
-->
<xsd:complexType name="fileExistTriggerType">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute name="comparison" type="comparisonFileExistTriggerType"
use="required"/>
      <xsd:attribute name="value" type="valueFileExistTriggerType" use="required"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
<!--
  Defines file size trigger type
  <fileSize comparison="=" value="1GB">file1,file2,file3</trigger>
-->
<xsd:complexType name="fileSizeTriggerType">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute name="comparison" type="comparisonFileSizeTriggerType"
use="required"/>
      <xsd:attribute name="value" type="valueFileSizeTriggerType" use="required"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
<!--
  Defines the enumeration values for file exists trigger conditions

```

```

        valueFileExistTriggerType="exist|noexist"
-->
<xsd:simpleType name="valueFileExistTriggerType">
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="exist"/>
    <xsd:enumeration value="noexist"/>
  </xsd:restriction>
</xsd:simpleType>
<!--
  Defines the enumeration values for file exists trigger comparison operator
  comparisonFileExistTriggerType="="|"!="
-->
<xsd:simpleType name="comparisonFileExistTriggerType">
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="=" />
    <xsd:enumeration value="!=" />
  </xsd:restriction>
</xsd:simpleType>
<!--
  Defines the enumeration values for file size trigger comparison operator
  comparisonFileSizeTriggerType=">="
-->
<xsd:simpleType name="comparisonFileSizeTriggerType">
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="&gt;=" />
  </xsd:restriction>
</xsd:simpleType>
<!--
  Defines the file size value pattern
  <fileSize comparison=">=" value="10|10B|10KB|10MB|10GB">file1</fileSize>
-->
<xsd:simpleType name="valueFileSizeTriggerType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[0123456789]+([bB]|[kK][bB]|[mM][bB]|[gG][bB]|)"/>
  </xsd:restriction>
</xsd:simpleType>
<!--
  Defines the enumeration values for trigger logging enabled flag
  <trigger log="yes|no">
-->
<xsd:simpleType name="logEnabledType">
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="yes"/>
    <xsd:enumeration value="no"/>
  </xsd:restriction>
</xsd:simpleType>
<!--
  Defines the reply type
  <reply QMGR="QMGR name" persistent="true">Queue Name</reply>
-->
<xsd:complexType name="replyType">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute name="QMGR" type="xsd:string" use="required"/>
      <xsd:attribute name="persistent" type="persistenceType" use="optional"/>
      <xsd:attribute name="detailed" type="detailedType"
use="optional"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>

<!--
  Defines the accepted choices for the detailed attribute.
-->
<xsd:simpleType name="detailedType">
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="true"/>
    <xsd:enumeration value="false"/>
  </xsd:restriction>
</xsd:simpleType>

<!--
  Defines the priority type
  <transferset priority="1">
-->
<xsd:simpleType name="priorityType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[0123456789]"/>
  </xsd:restriction>
</xsd:simpleType>
<!--
  Define the job information element

```

```

    <job>
      <name>JOBNAME</name>
    </job>
  -->
<xsd:complexType name="jobType">
  <xsd:sequence>
    <xsd:element name="name" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
<!--
  Defines an action
  <action>
    <runCommand name="myCommand.sh"/>
  </action>
  -->
<xsd:complexType name="commandActionType">
  <xsd:choice>
    <xsd:element name="command" type="commandType" maxOccurs="1" minOccurs="0"/>
  </xsd:choice>
</xsd:complexType>
<!--
  Defines a command
  <command name="runme" successRC="0" maxReplyLength="1024">
    <argument>firstArg</argument>
    <argument>secondArg</argument>
  </command>
  -->
<xsd:complexType name="commandType">
  <xsd:sequence>
    <xsd:element name="argument" type="xsd:string" maxOccurs="unbounded" minOccurs="0"/>
    <xsd:element name="target" type="xsd:string" maxOccurs="unbounded" minOccurs="0"/>
    <xsd:element name="property" type="propertyType" maxOccurs="unbounded"
minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="name" type="xsd:string" use="required"/>
  <xsd:attribute name="successRC" type="xsd:string" use="optional"/>
  <xsd:attribute name="retryCount" type="nonNegativeIntegerType" use="optional"/>
  <xsd:attribute name="retryWait" type="nonNegativeIntegerType" use="optional"/>
  <xsd:attribute name="type" type="callTypeType" use="optional"/>
  <xsd:attribute name="priority" type="commandPriorityType" use="optional"/>
  <xsd:attribute name="message" type="xsd:string" use="optional"/>
</xsd:complexType>
<!--
  Defines the enumeration values for the type of a command
  type="executable|antscript|jcl"
  -->
<xsd:simpleType name="callTypeType">
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="executable"/>
    <xsd:enumeration value="antscript"/>
    <xsd:enumeration value="jcl"/>
  </xsd:restriction>
</xsd:simpleType>
<!--
  Defines the priority type for a command
  priority="5"
  -->
<xsd:simpleType name="commandPriorityType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[123456789]"/>
  </xsd:restriction>
</xsd:simpleType>
<!--
  Defines the property type that is used as a child of commandType
  <property name="xxx" value="yyy"/>
  -->
<xsd:complexType name="propertyType">
  <xsd:attribute name="name" type="xsd:string" use="required"/>
  <xsd:attribute name="value" type="xsd:string" use="required"/>
</xsd:complexType>
<!-- Defines a non-negative integer type -->
<xsd:simpleType name="nonNegativeIntegerType">
  <xsd:restriction base="xsd:integer">
    <xsd:minInclusive value="0"/>
  </xsd:restriction>
</xsd:simpleType>
<!--
  Defines the transfer command specific reply information, to be included as part the
  general reply
  <transferReply>
    <preSourceData>

```

```

        <runCommandReply resultCode="0">
        <stdout>
            <line>the quick brown fox jumped over the lazy dog</line>
        </stdout>
        <stderr></stderr>
        </runCommandReply>
    </preSourceData>
</transferReply>
-->
<xsd:complexType name="transferReplyType">
    <xsd:sequence>
        <xsd:element name="preSourceData" type="actionReplyType" minOccurs="0"
maxOccurs="1"/>
        <xsd:element name="postSourceData" type="actionReplyType" minOccurs="0"
maxOccurs="1"/>
        <xsd:element name="preDestinationData" type="actionReplyType" minOccurs="0"
maxOccurs="1"/>
        <xsd:element name="postDestinationData" type="actionReplyType" minOccurs="0"
maxOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>
<!--
    Define the action reply type information
    <actionReply>
        <runCommandReply resultCode="1">
            <stdout></stdout>
            <stderr>
                <line>permission denied</line>
            </stderr>
        </runCommandReply>
    </actionReply>
-->
<xsd:complexType name="actionReplyType">
    <xsd:choice>
        <xsd:element name="runCommandReply" type="commandReplyType" maxOccurs="1"
minOccurs="0"/>
    </xsd:choice>
</xsd:complexType>
<!--
    Defines command specific reply information, to be included as part the general reply
    <commandReply resultCode="0">
        <stdout>
            <line>first line of output text</line>
            <line>second line of output text</line>
        </stdout>
        <stderr>
            <line>line of error text</line>
        </stderr>
    </commandReply>
-->
<xsd:complexType name="commandReplyType">
    <xsd:sequence>
        <xsd:element name="stdout" type="textLinesType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="stderr" type="textLinesType" maxOccurs="1" minOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute name="resultCode" type="xsd:int" use="required"/>
</xsd:complexType>
<!-- Defines type for lines of text -->
<xsd:complexType name="textLinesType">
    <xsd:sequence>
        <xsd:element name="line" type="xsd:string" maxOccurs="unbounded" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
<!--
    Defines the ping agent command specific reply information, to be included as part the
general reply
    <pingAgentReply resultCode="0">
        <agentVersion>Build level: f000-20090408-1200</agentVersion>
    </pingAgentReply>
-->
<xsd:complexType name="pingAgentReplyType">
    <xsd:sequence>
        <xsd:element name="agentVersion" type="xsd:string" maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
<!--
    Defines sequence of exit elements
    <exit ...
    <exit ...
-->
<xsd:complexType name="exitGroupType">
    <xsd:sequence>

```

```

        <xsd:element name="exit" type="exitType" minOccurs="1" maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
<!--
    Defines the outcome of calling a command
    <command ...
    <callResult ...
-->
<xsd:complexType name="callGroupType">
    <xsd:sequence>
        <xsd:element name="command" type="commandType" minOccurs="1" maxOccurs="1"/>
        <xsd:element name="callResult" type="callResultType" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>
<!--
    Defines either the successful call of a command, or a failed attempt to call a command
    <callResultType outcome="success|failure|error" retries="X">
        <result ... />
    </callResultType>
-->
<xsd:complexType name="callResultType">
    <xsd:sequence>
        <xsd:element name="result" type="resultType" minOccurs="1" maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:attribute name="returnCode" type="xsd:integer" use="optional"/>
    <xsd:attribute name="retries" type="xsd:integer" use="optional"/>
    <xsd:attribute name="outcome" type="outcomeType" use="required"/>
</xsd:complexType>
<!--
    Defines the information recorded for the successful call of a command
    <result...>
        <stdout...
        <stderr...
        <error...
    </result...>
-->
<xsd:complexType name="resultType">
    <xsd:sequence>
        <xsd:element name="stdout" type="outputType" minOccurs="0" maxOccurs="1"/>
        <xsd:element name="stderr" type="outputType" minOccurs="0" maxOccurs="1"/>
        <xsd:element name="error" type="xsd:string" maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="returnCode" type="xsd:integer" use="optional"/>
    <xsd:attribute name="outcome" type="outcomeType" use="required"/>
    <xsd:attribute name="time" type="xsd:dateTime" use="required"/>
</xsd:complexType>
<!-- Enumeration of call outcomes - success, failure or error -->
<xsd:simpleType name="outcomeType">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="success"/>
        <xsd:enumeration value="failure"/>
        <xsd:enumeration value="error"/>
    </xsd:restriction>
</xsd:simpleType>
<!--
    Defines the information recorded for each line of standard output / standard error
    generated by calling a program
    <line>line 1</line>
    <line>line 2</line>
    etc.
-->
<xsd:complexType name="outputType">
    <xsd:sequence>
        <xsd:element name="line" type="xsd:string" maxOccurs="unbounded" minOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>
<!--
    Defines the information recorded for an unsuccessful program call.
-->
<xsd:complexType name="callFailedType">
    <xsd:simpleContent>
        <xsd:extension base="xsd:string"/>
    </xsd:simpleContent>
</xsd:complexType>
<!--
    Defines the exit type; records the transfer exit class name and a status message
    <exit name="class com.example.exit.StartExit">
        <status ...
    </exit>
-->
<xsd:complexType name="exitType">
    <xsd:sequence>

```

```

        <xsd:element name="status" type="exitStatusType" minOccurs="1" maxOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute name="name" type="xsd:string" use="required"/>
</xsd:complexType>
<!--
    Defines exit status to record whether exit voted to proceed or cancel transfer.
    <status resultCode="proceed">
        <supplement>go ahead</supplement>
    </status>
-->
<xsd:complexType name="exitStatusType">
    <xsd:sequence>
        <xsd:element name="supplement" type="xsd:string" maxOccurs="unbounded"
minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="resultCode" type="exitResultEnumType" use="optional"/>
</xsd:complexType>
<!--
    Defines the enumeration for transfer exit result values.
    <status resultCode="proceed">
-->
<xsd:simpleType name="exitResultEnumType">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="proceed"/>
        <xsd:enumeration value="cancelTransfer"/>
        <xsd:enumeration value="cancelTask"/>
    </xsd:restriction>
</xsd:simpleType>
</xsd:schema>

```

Remarque : Depuis la IBM MQ 9.0, Managed File Transfer ne prend pas en charge la passerelle Web ou les agents Web.

Concepts associés

«Formats de message XML utilisés par MFT», à la page 756

Managed File Transfer utilise des messages au format XML à diverses fins: pour commander un agent, pour consigner des informations sur les moniteurs, les planifications et les transferts et pour définir les informations utilisées pour la configuration. Structure logique des formats XML utilisés à ces fins et décrits par le schéma XML.

Référence des applications de développement MFT

Exécution de programmes avant ou après un transfert de fichiers

Exemples d'utilisation de `fteCreateTransfer` pour démarrer des programmes

Vous pouvez utiliser la commande **`fteCreateTransfer`** pour spécifier les programmes à exécuter avant ou après un transfert.

Outre l'utilisation de **`fteCreateTransfer`**, il existe d'autres façons d'appeler un programme avant ou après un transfert. Pour plus d'informations, voir la section «[Spécification des programmes à exécuter avec MFT](#)», à la page 256.

Tous ces exemples utilisent la syntaxe suivante pour spécifier un programme:

```
[type:]commandspec[, [retrycount][, [retrywait][, successrc]]]
```

Pour plus d'informations sur cette syntaxe, voir «[fteCreateTransfer: démarrer un nouveau transfert de fichier](#)», à la page 447.

Exécution d'un programme exécutable

L'exemple suivant spécifie un programme exécutable appelé `mycommand` et transmet deux arguments, `a` et `b`, au programme.

```
mycommand(a,b)
```

Pour exécuter ce programme sur l'agent source AGENT1 avant le démarrage du transfert, utilisez la commande suivante:

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -presrc mycommand(a,b)
destinationSpecification sourceSpecification
```

Exécution et relance d'un programme exécutable

L'exemple suivant spécifie un programme exécutable appelé `simple`, qui ne prend aucun argument. Une valeur de 1 est spécifiée pour `retrycount` et une valeur de 5 est spécifiée pour `retrywait`. Ces valeurs signifient que le programme sera retenté une fois s'il ne renvoie pas un code retour réussi, après une attente de cinq secondes. Aucune valeur n'étant spécifiée pour `successrc`, le seul code retour réussi est la valeur par défaut 0.

```
executable:simple,1,5
```

Pour exécuter ce programme sur l'agent source AGENT1 une fois le transfert terminé, utilisez la commande suivante:

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -postsrc executable:simple,1,5
destinationSpecification sourceSpecification
```

Exécution d'un script Ant et spécification de codes retour ayant abouti

L'exemple suivant spécifie un script Ant appelé `myscript` et transmet deux propriétés au script. Le script est exécuté à l'aide de la commande **fteAnt**. La valeur de `successrc` est définie sur `>2&<7&!5|0|14`, ce qui indique que les codes retour 0, 3, 4, 6 et 14 indiquent une réussite.

```
antscript:myscript(prop1=fred,prop2=bob),,,>2&<7&!5|0|14
```

Pour exécuter ce programme sur l'agent de destination AGENT2 avant le début du transfert, utilisez la commande suivante:

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -predst
"antscript:myscript(prop1=fred,prop2=bob),,,>2&<7&!5|0|14"destinationSpecification sourceSpecification
```

Exécution d'un script Ant et spécification des cibles à appeler

L'exemple suivant spécifie un script Ant appelé `script2` et deux cibles, `target1` et `target2`, à appeler. La propriété `prop1` est également transmise, avec la valeur `recmfm(F,B)`. La virgule (,) et les parenthèses de cette valeur sont échappées à l'aide d'une barre oblique inversée (\).

```
antscript:script2(target1,target2,prop1=recmfm\F\B\),,,>2&<7&!5|0|14
```

Pour exécuter ce programme sur l'agent de destination AGENT2 une fois le transfert terminé, utilisez la commande suivante:

```
fteCreateTransfer -sa AGENT1 -da AGENT2
-postdst "antscript:script2(target1,target2,prop1=recmfm\F\B\),,,>2&<7&!5|0|14"
destinationSpecification sourceSpecification
```

Utilisation de métadonnées dans un script Ant

Vous pouvez spécifier une tâche Ant comme n'importe lequel des appels suivants pour un transfert:

- Pré-source
- Post-source
- prédestination
- post-destination

Lorsque la tâche Ant est exécutée, les métadonnées utilisateur du transfert sont rendues disponibles à l'aide de variables d'environnement. Vous pouvez accéder à ces données à l'aide, par exemple, du code suivant:

```
<property environment="environment"/>
<echo>${environment.mymetadata}</echo>
```

où mymetadata est le nom de certaines métadonnées insérées dans le transfert.

Exécution d'un script JCL

L'exemple suivant indique un script JCL appelé ZOSBATCH. Une valeur de 3 est spécifiée pour `retrycount`, une valeur de 30 est spécifiée pour `retrywait` et une valeur de 0 est spécifiée pour `successrc`. Ces valeurs signifient que le script est retenté trois fois s'il ne renvoie pas un code retour de réussite de 0, avec une attente de trente secondes entre chaque tentative.

```
jcl:ZOSBATCH,3,30,0
```

où ZOSBATCH est un membre d'un PDS appelé MYSYS.JCL et le fichier `agent.properties` contient la ligne `commandPath=...:/'MYSYS.JCL':...`

Pour exécuter ce programme sur l'agent source AGENT1 une fois le transfert terminé, utilisez la commande suivante:

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -postsrc jcl:ZOSBATCH,3,30,0
destinationSpecification sourceSpecification
```

Concepts associés

«[Spécification des programmes à exécuter avec MFT](#)», à la page 256

Vous pouvez exécuter des programmes sur un système sur lequel un Managed File Transfer Agent est en cours d'exécution. Dans le cadre d'une demande de transfert de fichiers, vous pouvez spécifier un programme à exécuter avant ou après le début d'un transfert. En outre, vous pouvez démarrer un programme qui ne fait pas partie d'une demande de transfert de fichier en soumettant une demande d'appel géré.

Référence associée

«[fteCreateTransfer: démarrer un nouveau transfert de fichier](#)», à la page 447

La commande **fteCreateTransfer** crée et démarre un nouveau transfert de fichier à partir de la ligne de commande. Cette commande peut démarrer un transfert de fichier immédiatement, planifier un transfert de fichier à une heure et une date ultérieures, répéter un transfert planifié une ou plusieurs fois et déclencher un transfert de fichiers sous certaines conditions.

Utilisation d' Apache Ant avec MFT

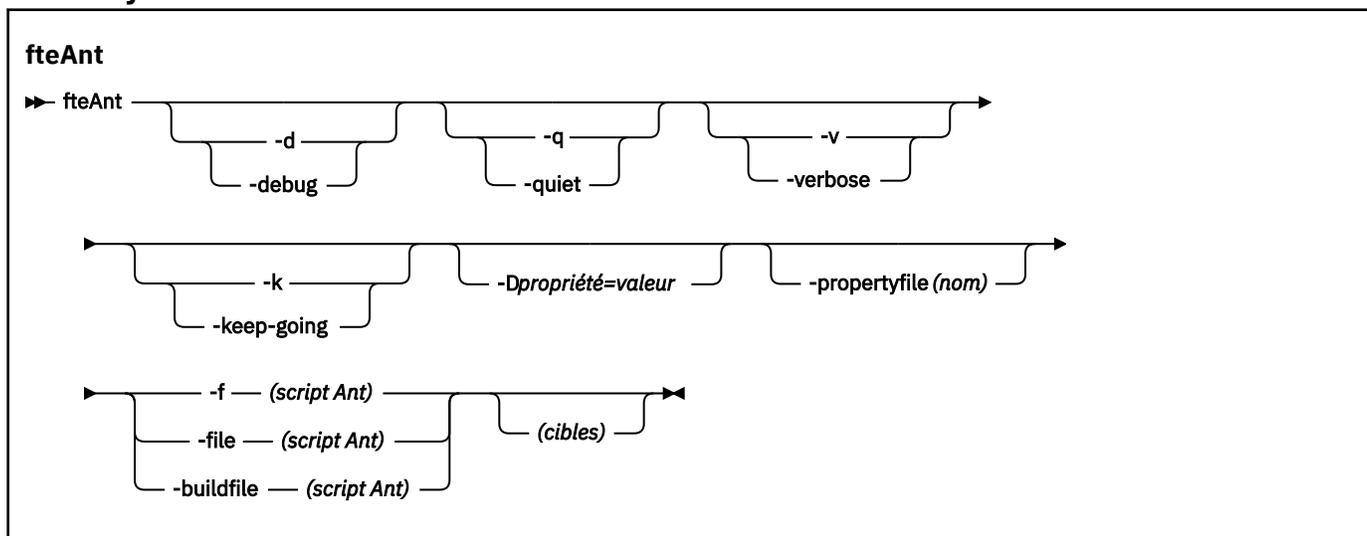
fteAnt: exécuter des tâches Ant dans MFT

La commande **fteAnt** exécute des scripts Ant dans un environnement dans lequel des tâches Ant Managed File Transfer sont disponibles.

Objet

Utilisez la commande **fteAnt** pour exécuter un script Ant dans un environnement avec Managed File Transfer. Contrairement à la commande **ant** standard, **fteAnt** requiert que vous définissiez un fichier script.

Syntaxe



Paramètres

-debug ou -d

Facultatif. Générez une sortie de débogage.

-quiet ou -q

Facultatif. Générer une sortie minimale.

-verbose ou -v

Facultatif. Générer une sortie prolixe.

-continue ou -k

Facultatif. Exécuter toutes les cibles qui ne dépendent pas des cibles ayant échoué.

-D propriété=valeur

Facultatif. Utilisez *value* pour une propriété *donnée*. Les propriétés définies avec **-D** sont prioritaires par rapport à celles définies dans un fichier de propriétés.

Utilisez la propriété **com.ibm.wmqfte.propertyset** pour spécifier l'ensemble des options de configuration utilisées pour les tâches Ant. Utilisez le nom d'un gestionnaire de files d'attente de coordination autre que celui par défaut comme valeur de cette propriété. Les tâches Ant utilisent ensuite l'ensemble des options de configuration associées à ce gestionnaire de files d'attente de coordination autre que celui par défaut. Si vous ne spécifiez pas cette propriété, l'ensemble d'options de configuration par défaut basé sur le gestionnaire de files d'attente de coordination par défaut est utilisé. Si vous spécifiez l'attribut **cmdqm** pour une tâche Ant, cet attribut est prioritaire sur l'ensemble des options de configuration spécifiées pour la commande **fteAnt**. Ce comportement s'applique que vous utilisiez l'ensemble d'options de configuration par défaut ou que vous spécifiez un ensemble avec la propriété **com.ibm.wmqfte.propertyset**.

-propertyfile (nom)

Facultatif. Chargez toutes les propriétés à partir d'un fichier dont les propriétés **-D** sont prioritaires.

-f (script Ant), -file (script Ant) ou -buildfile (script Ant)

Obligatoire. Indique le nom du script Ant à exécuter.

cibles

Facultatif. Nom d'une ou de plusieurs cibles à exécuter à partir du script Ant. Si vous ne spécifiez pas de valeur pour ce paramètre, la cible par défaut du script est exécutée.

-version

Facultatif. Affiche la commande Managed File Transfer et les versions Ant.

-? ou -h

Facultatif. Affiche la syntaxe de la commande.

Exemple

Dans cet exemple, la cible **copy** dans le script Ant `fte_script.xml` est exécutée et la commande écrit la sortie de débogage dans la sortie standard.

```
fteAnt -d -f fte_script.xml copy
```

Codes retour

0

La commande a abouti.

1

La commande a échoué.

D'autres codes retour de statut peuvent également être spécifiés à partir de scripts Ant, par exemple à l'aide de la tâche d'échec Ant.

Pour plus d'informations, voir [Echec](#).

Tâches Ant fournies par MFT

Managed File Transfer fournit un certain nombre de tâches Ant que vous pouvez utiliser pour accéder aux fonctions de transfert de fichiers.

Tâches

- [«fte: tâche Ant awaitissue», à la page 873](#)
- [fte: appel](#)
- [fte: annuler](#)
- [fte: filecopy](#)
- [fte: émoticône](#)
- [fte: ignoreissue](#)
- [fte: ping](#)
- [fte: uuid](#)

paramètres imbriqués

Les paramètres imbriqués suivants décrivent des ensembles imbriqués d'éléments, qui sont communs à plusieurs des tâches Ant fournies:

- [fte: spéc_fichier](#)
- [fte:metadata](#)
- [Paramètres d'appel de programme](#)

fte: tâche Ant awaitissue

Attendez la fin d'une opération **fte:filecopy**, **fte:filemove** ou **fte:call**.

Attribut

id

Obligatoire. Identifie le transfert à partir duquel attendre un résultat. En règle générale, il s'agit d'une propriété définie par l'attribut idProperty des tâches `fte:filecopy`, `fte:filemove` ou `fte:call`.

rcproperty

Obligatoire. Nomme une propriété dans laquelle stocker le code retour de la tâche `fte:awaitoutcome`.

dépassement du délai d'attente

Facultatif. Durée maximale, en secondes, d'attente de la fin de l'opération. Le délai d'attente minimal est d'une seconde. Si vous ne spécifiez pas de valeur de délai d'attente, la tâche `fte:awaitoutcome` attend indéfiniment que le résultat de l'opération soit déterminé.

Exemple

Dans cet exemple, une copie de fichier est démarrée et son identificateur est stocké dans la propriété `copy.id`. Pendant la progression de la copie, un autre traitement peut être effectué. L'instruction `fte:awaitoutcome` est utilisée pour attendre la fin de l'opération de copie. L'instruction `fte:awaitoutcome` identifie l'opération à attendre pour utiliser l'identificateur stocké dans la propriété `copy.id`. `fte:awaitoutcome` stocke un code retour indiquant le résultat de l'opération de copie dans une propriété appelée `copy.result`.

```
<!-- issue a file copy request -->
<fte:filecopy
  src="AGENT1@QM1"
  dst="AGENT2@QM2"
  idproperty="copy.id"
  outcome="defer">

  <fte:filespec
    srcfilespec="/home/fteuser1/file.bin"
    dstdir="/home/fteuser2"/>

</fte:filecopy>

<fte:awaitoutcome id="${copy.id}" rcProperty="copy.rc"/>

<echo>Copy id=${copy.id} rc=${copy.rc}</echo>
```

Référence associée

«Utilisation d' Apache Ant avec MFT», à la page 258

Managed File Transfer fournit des tâches que vous pouvez utiliser pour intégrer la fonction de transfert de fichier dans l'outil Ant Apache.

«Tâches Ant fournies par MFT», à la page 873

Managed File Transfer fournit un certain nombre de tâches Ant que vous pouvez utiliser pour accéder aux fonctions de transfert de fichiers.

fte: appel d'une tâche Ant

Vous pouvez utiliser la tâche `fte:call` pour appeler à distance des scripts et des programmes.

Cette tâche vous permet d'envoyer une demande `fte:call` à un agent. L'agent traite cette demande en exécutant un script ou un programme et en renvoyant le résultat. Les commandes à appeler doivent être accessibles à l'agent. Vérifiez que la valeur de la propriété `commandPath` dans le fichier `agent.properties` inclut l'emplacement des commandes à appeler. Les informations de chemin spécifiées par l'élément imbriqué de commande doivent être relatives aux emplacements spécifiés par la propriété `commandPath`. Par défaut, `commandPath` est vide, de sorte que l'agent ne peut pas appeler de commandes. Pour en savoir davantage sur cette propriété, reportez-vous à la section «Propriété `commandPath` MFT», à la page 374.

Pour plus d'informations sur le fichier `agent.properties`, voir «Le fichier MFT `agent.properties`», à la page 564.

Attribut

Agent

Obligatoire. Indique l'agent sur lequel la demande **fte:call** doit être soumise. Spécifiez les informations d'agent sous la forme suivante: *agentname@qmgrname* , où *agentname* est le nom de l'agent et *qmgrname* est le nom du gestionnaire de files d'attente auquel cet agent est directement connecté.

cmdqm

Facultatif. Gestionnaire de files d'attente de commandes auquel soumettre la demande. Spécifiez ces informations sous la forme *qmgrname@host@port@channel*, où:

- *qmgrname* est le nom du gestionnaire de files d'attente
- *host* est le nom d'hôte facultatif du système sur lequel le gestionnaire de files d'attente est en cours d'exécution.
- *port* est le numéro de port facultatif sur lequel le gestionnaire de files d'attente est en mode écoute
- *channel* est le canal SVRCONN facultatif à utiliser

Si vous omettez les informations *host*, *port* ou *channel* pour le gestionnaire de files d'attente de commandes, les informations de connexion spécifiées dans le fichier `command.properties` sont utilisées. Pour plus d'informations, voir la section [«Le fichier MFT command.properties»](#), à la page 559.

Vous pouvez utiliser la propriété **com.ibm.wmqfte.propertySet** pour spécifier le fichier `command.properties` à utiliser. Pour plus d'informations, voir [com.ibm.wmqfte.propertySet](#).

Si vous n'utilisez pas l'attribut `cmdqm` , la tâche utilise par défaut la propriété `com.ibm.wmqfte.ant.commandQueueManager` , si cette propriété est définie. Si la propriété `com.ibm.wmqfte.ant.commandQueueManager` n'est pas définie, une tentative de connexion au gestionnaire de files d'attente par défaut, défini dans le fichier `command.properties` , est effectuée. Le format de la propriété `com.ibm.wmqfte.ant.commandQueueManager` est identique à l'attribut `cmdqm` , c'est-à-dire *qmgrname@host@port@channel*.

idproperty

Facultatif sauf si vous avez spécifié un `outcome` de `defer`. Indique le nom d'une propriété à laquelle affecter l'identificateur de transfert. Les identificateurs de transfert sont générés au moment où une demande de transfert est soumise et vous pouvez utiliser des identificateurs de transfert pour suivre la progression d'un transfert, diagnostiquer les problèmes liés à un transfert et annuler un transfert.

Vous ne pouvez pas spécifier cette propriété si vous avez également spécifié une propriété `outcome` de `ignore`. Toutefois, vous devez spécifier `idproperty` si vous avez également spécifié une propriété `outcome` de `defer`.

Nom de travail

Facultatif. Affecte un nom de travail à la demande **fte:call** . Vous pouvez utiliser des noms de travail pour créer des groupes logiques de transferts. Utilisez la tâche `«fte: tâche Ant uuid»`, à la page 887 pour générer des noms de travail pseudo-uniqes. Si vous n'utilisez pas l'attribut `jobname` , la tâche utilise par défaut la valeur de la propriété `com.ibm.wmqfte.ant.jobName` , si cette propriété est définie. Si vous ne définissez pas cette propriété, aucun nom de travail n'est associé à la demande **fte:call** .

utilisateur d'origine

Facultatif. Indique l'ID utilisateur d'origine à associer à la demande **fte:call** . Si vous n'utilisez pas l'attribut `origuser` , la tâche utilise par défaut l'ID utilisateur utilisé pour exécuter le script Ant.

résultat

Facultatif. Détermine si la tâche attend la fin de l'opération **fte:call** avant de renvoyer le contrôle au script Ant. Indiquez l'une des options suivantes :

attendre

La tâche attend la fin de l'opération **fte:call** avant de revenir. Lorsqu'un outcome de `await` est spécifié, l'attribut `idproperty` est facultatif.

defer

La tâche est renvoyée dès que la demande **fte:call** a été soumise et suppose que le résultat de l'opération d'appel est traité ultérieurement à l'aide des tâches `awaitrésultat` ou `ignoreissue`. Lorsqu'une valeur outcome de `defer` est spécifiée, l'attribut `idproperty` est obligatoire.

ignorer

Si le résultat de l'opération **fte:call** n'est pas important, vous pouvez spécifier la valeur `ignore`. La tâche est ensuite renvoyée dès que la demande **fte:call** a été soumise, sans allouer de ressources pour le suivi du résultat de la commande. Lorsqu'une valeur outcome de `ignore` est spécifiée, l'attribut `idproperty` ne peut pas être spécifié.

Si vous ne spécifiez pas l'attribut `issue`, la tâche utilise par défaut la valeur `await`.

rcproperty

Facultatif. Indique le nom d'une propriété à laquelle affecter le code de résultat de la demande **fte:call**. Le code de résultat reflète le résultat global de la demande **fte:call**.

Vous ne pouvez pas spécifier cette propriété si vous avez également spécifié une propriété outcome `ignore` ou `defer`. Toutefois, vous devez spécifier `rcproperty` si vous avez spécifié un résultat de `await`.

Paramètres spécifiés en tant qu'éléments imbriqués

fte:commande

Indique la commande à appeler par l'agent. Vous ne pouvez associer qu'un seul élément `fte:command` à une opération **fte:call** donnée. La commande à appeler doit se trouver dans le chemin spécifié par la propriété `commandPath` dans le fichier `agent.properties` de l'agent.

fte:metadata

Vous pouvez spécifier des métadonnées à associer à l'opération d'appel. Ces métadonnées sont enregistrées dans les messages de journal générés par l'opération d'appel. Vous ne pouvez associer qu'un seul bloc de métadonnées à un élément de transfert donné ; cependant, ce bloc peut contenir de nombreux éléments de métadonnées.

Exemple

Cet exemple montre comment appeler une commande à l'adresse `AGENT1` s'exécutant sur le gestionnaire de files d'attente `QM1`. La commande à appeler est le script `command.sh`, et le script est appelé avec un seul argument `xyz`. La commande `command.sh` se trouve dans le chemin spécifié par la propriété `commandPath` dans le fichier `agent.properties` de l'agent.

```
<fte:call cmdqm="QM0@localhost@1414@SYSTEM.DEF.SVRCONN"
  agent="AGENT1@QM1"
  rcproperty="call.rc"
  origuser="bob"
  jobname="${job.id}">
  <fte:command command="command.sh" successrc="1" retrycount="5" retrywait="30">
    <fte:arg value="xyz"/>
  </fte:command>
  <fte:metadata>
    <fte:entry name="org.foo.accountName" value="BDG3R"/>
  </fte:metadata>
</fte:call>
```

Référence associée

«Utilisation d' Apache Ant avec MFT», à la page 258

Managed File Transfer fournit des tâches que vous pouvez utiliser pour intégrer la fonction de transfert de fichier dans l'outil Ant Apache.

«Tâches Ant fournies par MFT», à la page 873

Managed File Transfer fournit un certain nombre de tâches Ant que vous pouvez utiliser pour accéder aux fonctions de transfert de fichiers.

fte: annuler la tâche Ant

Annule un transfert géré ou un appel géré Managed File Transfer . Un transfert géré peut avoir été créé à l'aide des tâches **fte:filecopy** ou **fte:filemove** . Un appel géré a peut-être été créé à l'aide de la tâche **fte:call** .

Attribut

Agent

Obligatoire. Indique l'agent sur lequel la demande **fte:cancel** doit être soumise. La valeur se présente sous la forme suivante: *agentname@qmgrname* , où *agentname* est le nom de l'agent et *qmgrname* est le nom du gestionnaire de files d'attente auquel cet agent est directement connecté.

cmdqm

Facultatif. Gestionnaire de files d'attente de commandes auquel soumettre la demande. Spécifiez ces informations sous la forme *qmgrname@host@port@channel*, où:

- *qmgrname* est le nom du gestionnaire de files d'attente
- *host* est le nom d'hôte facultatif du système sur lequel le gestionnaire de files d'attente est en cours d'exécution.
- *port* est le numéro de port facultatif sur lequel le gestionnaire de files d'attente est en mode écoute
- *channel* est le canal SVRCONN facultatif à utiliser

Si vous omettez les informations *host*, *port* ou *channel* pour le gestionnaire de files d'attente de commandes, les informations de connexion spécifiées dans le fichier `command.properties` sont utilisées. Pour plus d'informations, voir la section «Le fichier MFT `command.properties`», à la page 559.

Vous pouvez utiliser la propriété **com.ibm.wmqfte.propertySet** pour spécifier le fichier `command.properties` à utiliser. Pour plus d'informations, voir [com.ibm.wmqfte.propertySet](#).

Si vous n'utilisez pas l'attribut `cmdqm` , la tâche utilise par défaut la propriété `com.ibm.wmqfte.ant.commandQueueManager` , si cette propriété est définie. Si la propriété `com.ibm.wmqfte.ant.commandQueueManager` n'est pas définie, une tentative de connexion au gestionnaire de files d'attente par défaut, défini dans le fichier `command.properties` , est effectuée. Le format de la propriété `com.ibm.wmqfte.ant.commandQueueManager` est identique à l'attribut `cmdqm` , c'est-à-dire *qmgrname@host@port@channel*.

id

Obligatoire. Indique l'identificateur de transfert du transfert à annuler. Les identificateurs de transfert sont générés au moment où une demande de transfert est soumise par les tâches [fte:filecopy](#) et [fte:filemove](#) .

utilisateur d'origine

Facultatif. Indique l'ID utilisateur d'origine à associer à la demande **cancel** . Si l'attribut `origuser` n'est pas utilisé, la tâche utilise par défaut l'ID utilisateur utilisé pour exécuter le script Ant.

Exemple

L'exemple envoie une demande **fte:cancel** au gestionnaire de files d'attente de commandes `qm0`. La demande **fte:cancel** est ciblée sur `agent1` sur le gestionnaire de files d'attente `qm1` pour l'identificateur de transfert renseigné par la variable `transfer.id` . La demande est exécutée à l'aide de l'ID utilisateur "bob" .

```
<fte:cancel cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  agent="agent1@qm1"
  id="{transfer.id}"
  origuser="bob"/>
```

Référence associée

«Utilisation d' Apache Ant avec MFT», à la page 258

Managed File Transfer fournit des tâches que vous pouvez utiliser pour intégrer la fonction de transfert de fichier dans l'outil Ant Apache .

«Tâches Ant fournies par MFT», à la page 873

Managed File Transfer fournit un certain nombre de tâches Ant que vous pouvez utiliser pour accéder aux fonctions de transfert de fichiers.

fte: tâche Ant filecopy

La tâche **fte:filecopy** copie les fichiers entre les agents Managed File Transfer . Le fichier n'est pas supprimé de l'agent source.

Attribut

cmdqm

Facultatif. Gestionnaire de files d'attente de commandes auquel soumettre la demande. Spécifiez ces informations sous la forme `qmgrname@host@port@channel`, où:

- `qmgrname` est le nom du gestionnaire de files d'attente
- `host` est le nom d'hôte facultatif du système sur lequel le gestionnaire de files d'attente est en cours d'exécution.
- `port` est le numéro de port facultatif sur lequel le gestionnaire de files d'attente est en mode écoute
- `channel` est le canal SVRCONN facultatif à utiliser

Si vous omettez les informations `host`, `port` ou `channel` pour le gestionnaire de files d'attente de commandes, les informations de connexion spécifiées dans le fichier `command.properties` sont utilisées. Pour plus d'informations, voir la section «Le fichier MFT `command.properties`», à la page 559.

Vous pouvez utiliser la propriété **com.ibm.wmqfte.propertySet** pour spécifier le fichier `command.properties` à utiliser. Pour plus d'informations, voir [com.ibm.wmqfte.propertySet](#).

Si vous n'utilisez pas l'attribut `cmdqm`, la tâche utilise par défaut la propriété `com.ibm.wmqfte.ant.commandQueueManager`, si cette propriété est définie. Si la propriété `com.ibm.wmqfte.ant.commandQueueManager` n'est pas définie, une tentative de connexion au gestionnaire de files d'attente par défaut, défini dans le fichier `command.properties`, est effectuée. Le format de la propriété `com.ibm.wmqfte.ant.commandQueueManager` est identique à l'attribut `cmdqm`, c'est-à-dire `qmgrname@host@port@channel`.

Outils de maintenance en mode dédié

Obligatoire. Indique l'agent de destination pour l'opération de copie. Spécifiez ces informations sous la forme suivante: `agentname@qmgrname` où `agentname` est le nom de l'agent de destination et `qmgrname` est le nom du gestionnaire de files d'attente auquel cet agent est directement connecté.

idproperty

Facultatif sauf si vous avez spécifié un `outcome` de `defer`. Indique le nom d'une propriété à laquelle affecter l'identificateur de transfert. Les identificateurs de transfert sont générés au moment où une demande de transfert est soumise et vous pouvez utiliser des identificateurs de transfert pour suivre la progression d'un transfert, diagnostiquer les problèmes liés à un transfert et annuler un transfert.

Vous ne pouvez pas spécifier cette propriété si vous avez également spécifié une propriété `outcome` de `ignore`. Toutefois, vous devez spécifier `idproperty` si vous avez également spécifié une propriété `outcome` de `defer`.

Nom de travail

Facultatif. Affecte un nom de travail à la demande de copie. Vous pouvez utiliser des noms de travail pour créer des groupes logiques de transferts. Utilisez la tâche «*fte: tâche Ant uuid*», à la page 887 pour générer des noms de travail pseudo-uniqes. Si vous n'utilisez pas l'attribut `jobname`, la tâche utilise par défaut la valeur de la propriété `com.ibm.wmqfte.ant.jobName`, si cette propriété est

définie. Si vous ne définissez pas cette propriété, aucun nom de travail n'est associé à la demande de copie.

utilisateur d'origine

Facultatif. Indique l'ID utilisateur d'origine à associer à la demande de copie. Si vous n'utilisez pas l'attribut `origuser`, la tâche utilise par défaut l'ID utilisateur utilisé pour exécuter le script Ant.

résultat

Facultatif. Détermine si la tâche attend la fin de l'opération de copie avant de renvoyer le contrôle au script Ant. Indiquez l'une des options suivantes :

attendre

La tâche attend la fin de l'opération de copie avant de revenir. Lorsqu'un outcome de `await` est spécifié, l'attribut `idproperty` est facultatif.

defer

La tâche est renvoyée dès que la demande de copie a été soumise et suppose que le résultat de l'opération de copie est traité ultérieurement à l'aide des tâches `awaitresultant` ou «`fte: tâche Ant ignoreissue`», à la page 885 . Lorsqu'une valeur outcome de `defer` est spécifiée, l'attribut `idproperty` est obligatoire.

ignorer

Si le résultat de l'opération de copie n'est pas important, vous pouvez spécifier la valeur `ignore`. La tâche revient ensuite dès que la demande de copie a été soumise, sans allouer de ressources pour suivre le résultat du transfert. Lorsqu'une valeur outcome de `ignore` est spécifiée, l'attribut `idproperty` ne peut pas être spécifié.

Si vous ne spécifiez pas l'attribut `issue`, la tâche utilise par défaut la valeur `await`.

priorité

Facultatif. Indique la priorité à associer à la demande de copie. En général, les demandes de transfert de priorité plus élevée sont prioritaires par rapport aux demandes de priorité plus faible. La valeur de priorité doit être comprise entre 0 et 9 (inclus). Une valeur de priorité de 0 correspond à la priorité la plus faible et une valeur de 9 à la priorité la plus élevée. Si vous ne spécifiez pas l'attribut `priority`, le transfert prend par défaut la valeur 0.

rcproperty

Facultatif. Indique le nom d'une propriété à laquelle affecter le code de résultat de la demande de copie. Le code de résultat reflète le résultat global de la demande de copie.

Vous ne pouvez pas spécifier cette propriété si vous avez également spécifié une propriété outcome `ignore` ou `defer`. Toutefois, vous devez spécifier `rcproperty` si vous spécifiez un résultat `await`.

V 9.0.1 transferRecoveryTimeout

Facultatif. Définit la durée, en secondes, pendant laquelle un agent source continue de tenter de récupérer un transfert de fichiers bloqué. Indiquez l'une des options suivantes :

-1

L'agent continue de tenter de reprendre le transfert bloqué jusqu'à ce que ce dernier soit terminé. Cette option est équivalente au comportement par défaut de l'agent si la propriété n'est pas définie.

0

L'agent arrête le transfert de fichiers dès qu'il entre dans la phase de reprise.

>0

L'agent continue de tenter de reprendre le transfert bloqué pendant le nombre de secondes défini par l'entier positif spécifié. Par exemple :

```
<fte:filecopy cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  src="agent1@qm1" dst="agent2@qm2"
  rcproperty="copy.result" transferRecoveryTimeout="21600">
<fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/
```

```
file.bin"/>
</fte:filecopy>
```

indique que l'agent tente de récupérer le transfert pendant 6 heures à partir du moment où il entre en récupération. La valeur maximale de cet attribut est 999999999.

La spécification de la valeur de délai d'attente de reprise de transfert de cette manière la définit par transfert. Pour définir une valeur globale pour tous les transferts dans un réseau Managed File Transfer, vous pouvez ajouter une propriété au fichier `agent.properties`. Pour plus d'informations, voir la section «[Option de délai d'attente pour les transferts de fichiers en cours de récupération](#)», à la page 253.

srce

Obligatoire. Indique l'agent source pour l'opération de copie. Spécifiez ces informations sous la forme suivante: `agentname@qmgrname`, où `agentname` est le nom de l'agent source et `qmgrname` est le nom du gestionnaire de files d'attente auquel cet agent est directement connecté.

Paramètres spécifiés en tant qu'éléments imbriqués

fte: spéc_fichier

Obligatoire. Vous devez spécifier au moins une spécification de fichier qui identifie les fichiers à copier. Vous pouvez spécifier plusieurs spécifications de fichier si nécessaire. Pour plus d'informations, voir la rubrique [fte: filespec](#).

fte: métadonnées

Vous pouvez spécifier des métadonnées à associer à l'opération de copie. Ces métadonnées sont transportées avec le transfert et sont enregistrées dans les messages de journal générés par le transfert. Vous ne pouvez associer qu'un seul bloc de métadonnées à un élément de transfert donné ; cependant, ce bloc peut contenir de nombreux éléments de métadonnées. Pour plus d'informations, voir la rubrique [fte: metadata](#).

fte: presrc

Indique un appel de programme à effectuer au niveau de l'agent source avant le début du transfert. Vous ne pouvez associer qu'un seul élément `fte: presrc` à un transfert donné. Pour plus d'informations, voir la rubrique [Appel de programme](#).

fte: predst

Indique un appel de programme à effectuer au niveau de l'agent de destination avant le début du transfert. Vous ne pouvez associer qu'un seul élément `fte: predst` à un transfert donné. Pour plus d'informations, voir la rubrique [Appel de programme](#).

fte: postsrc

Indique un appel de programme à effectuer au niveau de l'agent source une fois le transfert terminé. Vous ne pouvez associer qu'un seul élément `fte: postsrc` à un transfert donné. Pour plus d'informations, voir la rubrique [Appel de programme](#).

fte: postdst

Indique un appel de programme à effectuer au niveau de l'agent de destination une fois le transfert terminé. Vous ne pouvez associer qu'un seul élément `fte: postdst` à un transfert donné. Pour plus d'informations, voir la rubrique [Appel de programme](#).

Si `fte:presrc`, `fte:predst`, `fte:postsrc`, `fte:postdst` et que les exits ne renvoient pas de statut de réussite, les règles sont les suivantes dans l'ordre indiqué:

1. Exécutez les exits de démarrage de la source. Si les exits de démarrage de la source échouent, le transfert échoue et aucune autre opération n'est effectuée.
2. Exécutez l'appel présorce (s'il est présent). Si l'appel de présorce échoue, le transfert échoue et rien n'est exécuté.
3. Exécutez les exits de démarrage de destination. Si les exits de démarrage de la destination échouent, le transfert échoue et aucune autre opération n'est effectuée.
4. Exécutez l'appel de prédestination (s'il est présent). Si l'appel de prédestination échoue, le transfert échoue et rien n'est exécuté.

5. Effectuez les transferts de fichiers.
6. Exécutez les exits de fin de destination. Il n'y a pas de statut d'échec pour ces exits.
7. Si le transfert aboutit (si le transfert de certains fichiers aboutit, il est considéré comme ayant abouti), exécutez l'appel post-destination (s'il est présent). Si l'appel post-destination échoue, le transfert échoue.
8. Exécutez les exits d'extrémité source. Il n'y a pas de statut d'échec pour ces exits.
9. Si le transfert aboutit, exécutez l'appel post-source (s'il est présent). Si l'appel post-source échoue, le transfert échoue.

Exemples

Cet exemple illustre un transfert de fichier de base entre agent1 et agent2. La commande permettant de démarrer le transfert de fichiers est envoyée à un gestionnaire de files d'attente appelé qm0, à l'aide d'une connexion en mode transport client. Le résultat de l'opération de transfert de fichier est affecté à la propriété appelée copy.result.

```
<fte:filecopy cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  src="agent1@qm1" dst="agent2@qm2"
  rcproperty="copy.result">
  <fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/file.bin"/>
</fte:filecopy>
```

Cet exemple montre le même transfert de fichiers, mais avec l'ajout de métadonnées et le démarrage d'un programme au niveau de l'agent source une fois le transfert terminé.

```
<fte:filecopy cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  src="agent1@qm1" dst="agent2@qm2"
  rcproperty="copy.result">
  <fte:metadata>
    <fte:entry name="org.example.departId" value="ACCOUNTS"/>
    <fte:entry name="org.example.batchGroup" value="A1"/>
  </fte:metadata>
  <fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/file.bin"/>
  <fte:postsrc command="/home/fteuser2/scripts/post.sh" successsrc="1" >
    <fte:arg value="/home/fteuser2/file.bin"/>
  </fte:postsrc>
</fte:filecopy>
```

Concepts associés

«Option de délai d'attente pour les transferts de fichiers en cours de récupération», à la page 253

Vous pouvez définir la durée, en secondes, pendant laquelle un agent source tente de récupérer un transfert de fichiers bloqué. Si le transfert n'aboutit pas lorsque l'agent atteint le délai d'attente pour l'intervalle entre les nouvelles tentatives, le transfert échoue.

Référence associée

«Utilisation d' Apache Ant avec MFT», à la page 258

Managed File Transfer fournit des tâches que vous pouvez utiliser pour intégrer la fonction de transfert de fichier dans l'outil Ant Apache .

«Tâches Ant fournies par MFT», à la page 873

Managed File Transfer fournit un certain nombre de tâches Ant que vous pouvez utiliser pour accéder aux fonctions de transfert de fichiers.

fte: tâche Ant filemove

La tâche **fte:filemove** déplace les fichiers entre les agents Managed File Transfer . Lorsqu'un fichier a été transféré avec succès de l'agent source vers l'agent cible, le fichier est supprimé de l'agent source.

Attribut

cmdqm

Facultatif. Gestionnaire de files d'attente de commandes auquel soumettre la demande. Spécifiez ces informations sous la forme *qmgrname@host@port@channel*, où:

- *qmgrname* est le nom du gestionnaire de files d'attente
- *host* est le nom d'hôte facultatif du système sur lequel le gestionnaire de files d'attente est en cours d'exécution.
- *port* est le numéro de port facultatif sur lequel le gestionnaire de files d'attente est en mode écoute
- *channel* est le canal SVRCONN facultatif à utiliser

Si vous omettez les informations *host*, *port* ou *channel* pour le gestionnaire de files d'attente de commandes, les informations de connexion spécifiées dans le fichier `command.properties` sont utilisées. Pour plus d'informations, voir la section «Le fichier MFT `command.properties`», à la page 559.

Vous pouvez utiliser la propriété **com.ibm.wmqfte.propertySet** pour spécifier le fichier `command.properties` à utiliser. Pour plus d'informations, voir [com.ibm.wmqfte.propertySet](#).

Si vous n'utilisez pas l'attribut `cmdqm`, la tâche utilise par défaut la propriété `com.ibm.wmqfte.ant.commandQueueManager`, si cette propriété est définie. Si la propriété `com.ibm.wmqfte.ant.commandQueueManager` n'est pas définie, une tentative de connexion au gestionnaire de files d'attente par défaut, défini dans le fichier `command.properties`, est effectuée. Le format de la propriété `com.ibm.wmqfte.ant.commandQueueManager` est identique à l'attribut `cmdqm`, c'est-à-dire *qmgrname@host@port@channel*.

Outils de maintenance en mode dédié

Obligatoire. Indique l'agent de destination pour l'opération de copie. Spécifiez ces informations sous la forme suivante: *agentname@qmgrname* où *agentname* est le nom de l'agent de destination et *qmgrname* est le nom du gestionnaire de files d'attente auquel cet agent est directement connecté.

idproperty

Facultatif sauf si vous avez spécifié un `outcome` de `defer`. Indique le nom d'une propriété à laquelle affecter l'identificateur de transfert. Les identificateurs de transfert sont générés au moment où une demande de transfert est soumise et vous pouvez utiliser des identificateurs de transfert pour suivre la progression d'un transfert, diagnostiquer les problèmes liés à un transfert et annuler un transfert.

Vous ne pouvez pas spécifier cette propriété si vous avez également spécifié une propriété `outcome` de `ignore`. Toutefois, vous devez spécifier `idproperty` si vous avez également spécifié une propriété `outcome` de `defer`.

Nom de travail

Facultatif. Affecte un nom de travail à la demande de déplacement. Vous pouvez utiliser des noms de travail pour créer des groupes logiques de transferts. Utilisez la tâche `fte:uuid` pour générer des noms de travail pseudo-uniqes. Si vous n'utilisez pas l'attribut `jobname`, la tâche utilise par défaut la valeur de la propriété `com.ibm.wmqfte.ant.jobName`, si cette propriété est définie. Si vous ne définissez pas cette propriété, aucun nom de travail n'est associé à la demande de déplacement.

utilisateur d'origine

Facultatif. Indique l'ID utilisateur d'origine à associer à la demande de déplacement. Si vous n'utilisez pas l'attribut `origuser`, la tâche utilise par défaut l'ID utilisateur utilisé pour exécuter le script Ant.

résultat

Facultatif. Détermine si la tâche attend la fin de l'opération de déplacement avant de renvoyer le contrôle au script Ant. Indiquez l'une des options suivantes :

attendre

La tâche attend la fin de l'opération de déplacement avant de revenir. Lorsqu'un `outcome` de `await` est spécifié, l'attribut `idproperty` est facultatif.

defer

La tâche est renvoyée dès que la demande de déplacement a été soumise et suppose que le résultat de l'opération de déplacement est traité ultérieurement à l'aide de la tâche «fte: tâche Ant awaitissue», à la page 873 ou «fte: tâche Ant ignoreissue», à la page 885 . Lorsqu'une valeur outcome de defer est spécifiée, l'attribut idproperty est obligatoire.

ignorer

Si le résultat de l'opération de déplacement n'est pas important, vous pouvez spécifier la valeur ignore. La tâche revient ensuite dès que la demande de déplacement a été soumise, sans allouer de ressources pour le suivi du résultat du transfert. Lorsqu'une valeur outcome de ignore est spécifiée, l'attribut idproperty ne peut pas être spécifié.

Si vous ne spécifiez pas l'attribut issue , la tâche utilise par défaut la valeur await.

priorité

Facultatif. Indique la priorité à associer à la demande de déplacement. En général, les demandes de transfert de priorité plus élevée sont prioritaires par rapport aux demandes de priorité plus faible. La valeur de priorité doit être comprise entre 0 et 9 (inclus). Une valeur de priorité de 0 correspond à la priorité la plus faible et une valeur de 9 à la priorité la plus élevée. Si vous ne spécifiez pas l'attribut priority , le transfert prend par défaut la valeur 0.

rcproperty

Facultatif. Indique le nom d'une propriété à laquelle affecter le code de résultat de la demande de déplacement. Le code de résultat reflète le résultat global de la demande de déplacement.

Vous ne pouvez pas spécifier cette propriété si vous avez également spécifié une propriété outcome ignore ou defer. Toutefois, vous devez spécifier rcproperty si vous avez spécifié un résultat de await.

V 9.01 transferRecoveryTimeout

Facultatif. Définit la durée, en secondes, pendant laquelle un agent source continue de tenter de récupérer un transfert de fichiers bloqué. Indiquez l'une des options suivantes :

-1

L'agent continue de tenter de reprendre le transfert bloqué jusqu'à ce que ce dernier soit terminé. Cette option est équivalente au comportement par défaut de l'agent si la propriété n'est pas définie.

0

L'agent arrête le transfert de fichiers dès qu'il entre dans la phase de reprise.

>0

L'agent continue de tenter de reprendre le transfert bloqué pendant le nombre de secondes défini par l'entier positif spécifié. Par exemple :

```
<fte:filemove cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  src=agent1@qm1 dst="agent2@qm2"
  rcproperty="move.result" transferRecoveryTimeout="21600">
  <fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/
file.bin"/>
</fte:filemove
```

indique que l'agent tente de récupérer le transfert pendant 6 heures à partir du moment où il entre en récupération. La valeur maximale de cet attribut est 999999999.

La spécification de la valeur de délai d'attente de reprise de transfert de cette manière la définit par transfert. Pour définir une valeur globale pour tous les transferts dans un réseau Managed File Transfer , vous pouvez ajouter une propriété au fichier agent.properties. Pour plus d'informations, voir la section «[Option de délai d'attente pour les transferts de fichiers en cours de récupération](#)», à la page 253.

srce

Obligatoire. Indique l'agent source pour l'opération de déplacement. Spécifiez ces informations sous la forme suivante: *agentname@qmgrname* où *agentname* est le nom de l'agent source et *qmgrname* est le nom du gestionnaire de files d'attente auquel cet agent est directement connecté.

Paramètres spécifiés en tant qu'éléments imbriqués**fte: spéc_fichier**

Obligatoire. Vous devez spécifier au moins une spécification de fichier qui identifie les fichiers à déplacer. Vous pouvez spécifier plusieurs spécifications de fichier si nécessaire. Pour plus d'informations, voir la rubrique [fte: filespec](#) .

fte: métadonnées

Facultatif. Vous pouvez spécifier des métadonnées à associer à l'opération de déplacement de fichier. Ces métadonnées sont transportées avec le transfert et sont enregistrées dans les messages de journal générés par le transfert. Vous ne pouvez associer qu'un seul bloc de métadonnées à un élément de transfert donné ; cependant, ce bloc peut contenir de nombreux éléments de métadonnées. Pour plus d'informations, voir la rubrique [fte: metadata](#) .

fte: presrc

Facultatif. Indique un appel de programme à effectuer au niveau de l'agent source avant le début du transfert. Vous ne pouvez associer qu'un seul élément `fte: presrc` à un transfert donné. Pour plus d'informations, voir la rubrique [Appel de programme](#) .

fte: predst

Facultatif. Indique un appel de programme à effectuer au niveau de l'agent de destination avant le début du transfert. Vous ne pouvez associer qu'un seul élément `fte: predst` à un transfert donné. Pour plus d'informations, voir la rubrique [Appel de programme](#) .

fte: postsrc

Facultatif. Indique un appel de programme à effectuer au niveau de l'agent source une fois le transfert terminé. Vous ne pouvez associer qu'un seul élément `fte: postsrc` à un transfert donné. Pour plus d'informations, voir la rubrique [Appel de programme](#) .

fte: postdst

Facultatif. Indique un appel de programme à effectuer au niveau de l'agent de destination une fois le transfert terminé. Vous ne pouvez associer qu'un seul élément `fte: postdst` à un transfert donné. Pour plus d'informations, voir la rubrique [Appel de programme](#) .

Si `fte:presrc`, `fte:predst`, `fte:postsrc`, `fte:postdst` et que les exits ne renvoient pas de statut de réussite, les règles sont les suivantes dans l'ordre indiqué:

1. Exécutez les exits de démarrage de la source. Si les exits de démarrage de la source échouent, le transfert échoue et aucune autre opération n'est effectuée.
2. Exécutez l'appel présource (s'il est présent). Si l'appel de présource échoue, le transfert échoue et rien n'est exécuté.
3. Exécutez les exits de démarrage de destination. Si les exits de démarrage de la destination échouent, le transfert échoue et aucune autre opération n'est effectuée.
4. Exécutez l'appel de prédestination (s'il est présent). Si l'appel de prédestination échoue, le transfert échoue et rien n'est exécuté.
5. Effectuez les transferts de fichiers.
6. Exécutez les exits de fin de destination. Il n'y a pas de statut d'échec pour ces exits.
7. Si le transfert aboutit (si certains fichiers sont transférés avec succès, le transfert est considéré comme ayant abouti), exécutez l'appel post-destination (s'il est présent). Si l'appel post-destination échoue, le transfert échoue.
8. Exécutez les exits d'extrémité source. Il n'y a pas de statut d'échec pour ces exits.
9. Si le transfert aboutit, exécutez l'appel post-source (s'il est présent). Si l'appel post-source échoue, le transfert échoue.

Exemples

Cet exemple illustre un déplacement de fichier de base entre agent1 et agent2. La commande permettant de démarrer le déplacement de fichier est envoyée à un gestionnaire de files d'attente appelé qm0, à l'aide d'une connexion en mode transport client. Le résultat de l'opération de transfert de fichier est affecté à la propriété appelée `move.result`.

```
<fte:filemove cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  src="agent1@qm1" dst="agent2@qm2"
  rcproperty="move.result">

  <fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/file.bin"/>
</fte:filemove>
```

Concepts associés

«[Option de délai d'attente pour les transferts de fichiers en cours de récupération](#)», à la page 253
Vous pouvez définir la durée, en secondes, pendant laquelle un agent source tente de récupérer un transfert de fichiers bloqué. Si le transfert n'aboutit pas lorsque l'agent atteint le délai d'attente pour l'intervalle entre les nouvelles tentatives, le transfert échoue.

Référence associée

«[Utilisation d' Apache Ant avec MFT](#)», à la page 258
Managed File Transfer fournit des tâches que vous pouvez utiliser pour intégrer la fonction de transfert de fichier dans l'outil Ant Apache .

«[Tâches Ant fournies par MFT](#)», à la page 873

Managed File Transfer fournit un certain nombre de tâches Ant que vous pouvez utiliser pour accéder aux fonctions de transfert de fichiers.

fte: tâche Ant ignoreissue

Ignorez le résultat d'une commande **fte:filecopy**, **fte:filemove** ou **fte:call** . Lorsque vous spécifiez une tâche **fte:filecopy**, **fte:filemove** ou **fte:call** pour que le résultat soit `defer`, la tâche Ant alloue des ressources pour le suivi de ce résultat. Si le résultat ne vous intéresse plus, vous pouvez utiliser la tâche **fte:ignoreoutcome** pour libérer ces ressources.

Attribut

id

Obligatoire. Identifie le résultat qui n'a plus d'intérêt. En règle générale, vous spécifiez cet identificateur à l'aide d'une propriété que vous définissez à l'aide de l'attribut `idproperty` de la tâche «[fte: tâche Ant filecopy](#)», à la page 878, «[fte: tâche Ant filemove](#)», à la page 881 ou «[fte: appel d'une tâche Ant](#)», à la page 874 .

Exemple

Cet exemple montre comment utiliser la tâche `fte: ignoreissue` pour libérer les ressources allouées au suivi du résultat de la tâche «[fte: tâche Ant filecopy](#)», à la page 878 précédente.

```
<!-- issue a file copy request -->
<fte:filecopy cmdqm="qm1@localhost@1414@SYSTEM.DEF.SVRCONN"
  src="agent1@qm1" dst="agent1@qm1"
  idproperty="copy.id"
  outcome="defer"/>

<!-- do some other things -->

<!-- decide that the result of the copy is not interesting -->
<fte:ignoreoutcome id="${copy.id}"/>
```

Référence associée

«[Utilisation d' Apache Ant avec MFT](#)», à la page 258

Managed File Transfer fournit des tâches que vous pouvez utiliser pour intégrer la fonction de transfert de fichier dans l'outil Ant Apache .

«Tâches Ant fournies par MFT», à la page 873

Managed File Transfer fournit un certain nombre de tâches Ant que vous pouvez utiliser pour accéder aux fonctions de transfert de fichiers.

tâche Ant fte: ping

Cette tâche Ant IBM MQ Managed File Transfer envoie une commande ping à un agent pour obtenir une réponse et détermine ainsi si l'agent peut traiter les transferts.

Attribut

Agent

Obligatoire. Indique l'agent sur lequel la demande **fte:ping** doit être soumise. La valeur se présente sous la forme suivante: *agentname@qmgrname* , où *agentname* est le nom de l'agent et *qmgrname* est le nom du gestionnaire de files d'attente auquel cet agent est directement connecté.

cmdqm

Facultatif. Gestionnaire de files d'attente de commandes auquel soumettre la demande. Spécifiez ces informations sous la forme *qmgrname@host@port@channel*, où:

- *qmgrname* est le nom du gestionnaire de files d'attente
- *host* est le nom d'hôte facultatif du système sur lequel le gestionnaire de files d'attente est en cours d'exécution.
- *port* est le numéro de port facultatif sur lequel le gestionnaire de files d'attente est en mode écoute
- *channel* est le canal SVRCONN facultatif à utiliser

Si vous omettez les informations *host*, *port* ou *channel* pour le gestionnaire de files d'attente de commandes, les informations de connexion spécifiées dans le fichier `command.properties` sont utilisées. Pour plus d'informations, voir la section «Le fichier MFT `command.properties`», à la page 559.

Vous pouvez utiliser la propriété **com.ibm.wmqfte.propertySet** pour spécifier le fichier `command.properties` à utiliser. Pour plus d'informations, voir [com.ibm.wmqfte.propertySet](#).

Si vous n'utilisez pas l'attribut `cmdqm` , la tâche utilise par défaut la propriété `com.ibm.wmqfte.ant.commandQueueManager` , si cette propriété est définie. Si la propriété `com.ibm.wmqfte.ant.commandQueueManager` n'est pas définie, une tentative de connexion au gestionnaire de files d'attente par défaut, défini dans le fichier `command.properties` , est effectuée. Le format de la propriété `com.ibm.wmqfte.ant.commandQueueManager` est identique à l'attribut `cmdqm` , c'est-à-dire *qmgrname@host@port@channel*.

rcproperty

Obligatoire. Nomme une propriété dans laquelle stocker le code retour de l'opération **ping** .

dépassement du délai d'attente

Facultatif. Durée maximale, en secondes, pendant laquelle la tâche attend la réponse de l'agent. Le délai d'attente minimal est de zéro seconde, mais un délai d'attente de moins un peut également être spécifié de sorte que la commande attend indéfiniment que l'agent réponde. Si aucune valeur n'est spécifiée pour `timeout` , la valeur par défaut est d'attendre jusqu'à 5 secondes que l'agent réponde.

Exemple

Cet exemple envoie une demande **fte:ping** à `agent1` hébergé par `qm1`. La demande **fte:ping** attend 15 secondes que l'agent réponde. Le résultat de la demande **fte:ping** est stocké dans une propriété appelée `ping.rc`.

```
<fte:ping agent="agent1@qm1" rcproperty="ping.rc" timeout="15"/>
```

Codes retour

0

La commande a abouti.

2

La commande a expiré.

Référence associée

«Utilisation d' Apache Ant avec MFT», à la page 258

Managed File Transfer fournit des tâches que vous pouvez utiliser pour intégrer la fonction de transfert de fichier dans l'outil Ant Apache .

«Tâches Ant fournies par MFT», à la page 873

Managed File Transfer fournit un certain nombre de tâches Ant que vous pouvez utiliser pour accéder aux fonctions de transfert de fichiers.

fte: tâche Ant uuid

Génère un identificateur unique pseudo-aléatoire et l'affecte à une propriété donnée. Par exemple, vous pouvez utiliser cet identificateur pour générer des noms de travail pour d'autres opérations de transfert de fichiers.

Attribut

longueur

Obligatoire. Longueur numérique de l'identificateur unique universel à générer. Cette valeur de longueur n'inclut pas la longueur d'un préfixe, spécifiée par le paramètre **prefix** .

property

Obligatoire. Nom de la propriété à laquelle affecter l'identificateur unique universel généré.

préfixe

Facultatif. Préfixe à ajouter à l'identificateur unique universel généré. Ce préfixe n'est pas compté comme faisant partie de la longueur de l'UUID, comme indiqué par le paramètre **length** .

Exemple

Cet exemple définit un UUID qui commence par les lettres ABC suivies de 16 caractères hexadécimaux pseudo-aléatoires. L'identificateur unique universel est affecté à une propriété nommée `uuid.property`.

```
<fte:uuid length="16" property="uuid.property" prefix="ABC"/>
```

Référence associée

«Utilisation d' Apache Ant avec MFT», à la page 258

Managed File Transfer fournit des tâches que vous pouvez utiliser pour intégrer la fonction de transfert de fichier dans l'outil Ant Apache .

«Tâches Ant fournies par MFT», à la page 873

Managed File Transfer fournit un certain nombre de tâches Ant que vous pouvez utiliser pour accéder aux fonctions de transfert de fichiers.

fte: élément imbriqué Ant filespec

Le paramètre **fte:filespec** est utilisé en tant qu'élément imbriqué dans d'autres tâches. Utilisez

fte:filespec pour décrire un mappage entre un ou plusieurs fichiers source, répertoires  ou ensembles de données et une destination. Généralement, cet élément est utilisé lors de l'expression d'un ensemble de fichiers ou de répertoires  ou d'ensembles de données à déplacer ou à copier.

Imbriqué par:

- La tâche [fte: filecopy](#)
- Tâche [fte: filemove](#)

Attributs de spécification source

Vous devez spécifier `srcfilespec` ou `srcqueue`.

spéc_fichier_source

Indique la source de l'opération de fichier. La valeur de cet attribut peut inclure un caractère générique.

file_src

Indique que la source du transfert est une file d'attente. Le transfert déplace les données des messages stockés dans la file d'attente spécifiée par cet attribut. Vous ne pouvez pas spécifier cet attribut si la tâche **fte: filespec** est imbriquée dans la tâche **fte: filecopy**.

L'attribut `srcqueue` n'est pas pris en charge lorsque l'agent source est un agent de pont de protocole.

Attributs de spécification de destination

Vous devez spécifier `dstdir`, `dstds`, `dstfilespace`, `dstfile`, `dstqueue` ou `dstpds`.

dstdir

Indique un répertoire comme destination d'une opération de fichier.

dstds

Indique un fichier comme destination d'une opération de fichier.

Cet attribut est pris en charge uniquement lorsque l'agent de destination est en cours d'exécution sur la plateforme z/OS.

dstfile

Indique un fichier comme destination d'une opération de fichier.

espace fichier dstl

Indique un espace fichier comme destination d'une opération de fichier.

Cet attribut s'applique uniquement si l'agent de destination est un agent Web IBM MQ 8.0 qui a accès à l'espace fichier de la passerelle Web.

dstpds

Indique un fichier partitionné comme destination d'une opération de fichier.

Cet attribut est pris en charge uniquement lorsque l'agent de destination est en cours d'exécution sur la plateforme z/OS.

file d'attente de données

Indique une file d'attente comme destination d'une opération de fichier à message. Vous pouvez éventuellement inclure un nom de gestionnaire de files d'attente dans cette spécification, en utilisant le format `QUEUE@QUEUEMANAGER`. Si vous n'indiquez pas de nom de gestionnaire de files d'attente, le gestionnaire de files d'attente de l'agent de destination est utilisé si vous n'avez pas défini la propriété d'agent de sortie `enableClusterQueueInput` sur `true`. Si la propriété `enableClusterQueueInputOutput` est définie sur `true`, l'agent de destination utilise des procédures IBM MQ standard pour déterminer où se trouve la file d'attente. Vous devez indiquer un nom de file d'attente valide existant sur le gestionnaire de files d'attente.

Si vous spécifiez l'attribut `dstqueue`, vous ne pouvez pas spécifier les attributs `srcqueue` car ces attributs s'excluent mutuellement.

L'attribut `dstqueue` n'est pas pris en charge lorsque l'agent de destination est un agent de pont de protocole.

Attributs de l'option source

srcencoding

Facultatif. Codage de jeu de caractères utilisé par le fichier à transférer.

Vous pouvez spécifier cet attribut uniquement lorsque l'attribut `conversion` est défini sur la valeur `text`.

Si vous ne spécifiez pas l'attribut `srcencoding`, le jeu de caractères du système source est utilisé pour les transferts de texte.

srceol

Facultatif. Délimiteur de fin de ligne utilisé par le fichier en cours de transfert. Les valeurs admises sont les suivantes :

- `CRLF` -Utilisez un caractère de retour chariot suivi d'un caractère de saut de ligne comme délimiteur de fin de ligne. Cette convention est typique des systèmes Windows .
- `LF` -Utilisez un caractère de saut de ligne comme délimiteur de fin de ligne. Cette convention est typique des systèmes UNIX .

Vous pouvez spécifier cet attribut uniquement lorsque l'attribut `conversion` est défini sur la valeur `text`. Si vous ne spécifiez pas l'attribut `srceol`, les transferts de texte déterminent automatiquement la valeur correcte en fonction du système d'exploitation de l'agent source.

srckeeptrailingspaces

Facultatif. Détermine si les espaces de fin sont conservés dans les enregistrements source lus à partir d'un jeu de données de longueur fixe dans le cadre d'un transfert en mode texte. Les valeurs admises sont les suivantes :

- `true` -les espaces de fin sont conservés.
- `false` -les espaces de fin sont supprimés.

Si vous ne spécifiez pas l'attribut `srckeeptrailingspaces`, la valeur par défaut `false` est spécifiée.

Vous pouvez spécifier cet attribut uniquement si vous spécifiez également l'attribut `srcfilespec` et que vous affectez à l'attribut `conversion` la valeur `text`.

srcmsgdelimbytes

Facultatif. Indique une ou plusieurs valeurs d'octet à insérer comme délimiteur lors de l'ajout de plusieurs messages à un fichier binaire. Chaque valeur doit être spécifiée sous la forme de deux chiffres hexadécimaux dans la plage 00-FF, préfixés par `x`. Plusieurs octets doivent être séparés par des virgules. Par exemple, `srcmsgdelimbytes="x08,xA4"`. Vous ne pouvez spécifier l'attribut `srcmsgdelimbytes` que si vous avez également spécifié l'attribut `srcqueue`. Vous ne pouvez pas spécifier l'attribut `srcmsgdelimbytes` si vous avez également spécifié la valeur `text` pour l'attribut `conversion`.

srcmsgdelimtext

Facultatif. Indique une séquence de texte à insérer comme délimiteur lors de l'ajout de plusieurs messages à un fichier texte. Vous pouvez inclure des séquences d'échappement Java pour les littéraux chaîne dans le délimiteur. Par exemple, `srcmsgdelimtext="\u007d\n"`. Le délimiteur de texte est inséré après chaque message par l'agent source. Le délimiteur de texte est codé au format binaire à l'aide du codage source du transfert. Chaque message est lu au format binaire, le délimiteur codé est ajouté au message au format binaire et le résultat est transféré au format binaire à l'agent de destination. Si la page de codes de l'agent source inclut des états d'arrivée et de sortie, l'agent suppose que chaque message est à l'état de sortie à la fin du message. Au niveau de l'agent de destination, les données binaires sont converties de la même manière qu'un transfert de texte de fichier à fichier. Vous ne pouvez spécifier l'attribut `srcmsgdelimtext` que si vous avez également spécifié l'attribut `srcqueue` et la valeur `text` pour l'attribut `conversion`.

srcmsgdelimposition

Facultatif. Indique la position dans laquelle le texte ou le délimiteur binaire est inséré. Les valeurs admises sont les suivantes :

- **prefix** -les délimiteurs sont insérés dans le fichier de destination avant les données de chaque message.
- **postfix** -les délimiteurs sont insérés dans le fichier de destination après les données de chaque message.

Vous pouvez spécifier l'attribut `srcmsgdelimposition` uniquement si vous avez également spécifié l'un des attributs `srcmsgdelimbytes` ou `srcmsgdelimtext` .

srcmsggroups

Facultatif. Indique que les messages sont regroupés par ID groupe IBM MQ . Le premier groupe complet est écrit dans le fichier de destination. Si cet attribut n'est pas spécifié, tous les messages de la file d'attente source sont écrits dans le fichier de destination. Vous pouvez spécifier l'attribut `srcmsggroups` uniquement si vous avez également spécifié l'attribut `srcqueue` .

srcqueuetimeout

Facultatif. Indique le délai d'attente, en secondes, pour que l'une des conditions suivantes soit remplie:

- Pour qu'un nouveau message soit écrit dans la file d'attente.
- Si l'attribut `srcmsggroups` a été spécifié, pour qu'un groupe complet soit écrit dans la file d'attente.

Si aucune de ces conditions n'est remplie dans le délai spécifié par la valeur de `srcqueuetimeout`, l'agent source arrête la lecture à partir de la file d'attente et termine le transfert. Si l'attribut `srcqueuetimeout` n'est pas spécifié, l'agent source arrête de lire la file d'attente source immédiatement si la file d'attente source est vide ou, dans le cas où l'attribut `srcmsggroups` est spécifié, s'il n'y a pas de groupe complet dans la file d'attente. Vous pouvez spécifier l'attribut `srcqueuetimeout` uniquement si vous avez également spécifié l'attribut `srcqueue` .

Pour plus d'informations sur la définition de la valeur `srcqueuetimeout`, voir [«Conseils pour la spécification d'un temps d'attente sur un transfert de message vers fichier»](#), à la page 704.

z/OS srcrcdelimbytes

Facultatif. Indique une ou plusieurs valeurs d'octet à insérer comme délimiteur lors de l'ajout de plusieurs enregistrements d'un fichier source orienté enregistrement à un fichier binaire. Vous devez spécifier chaque valeur sous la forme de deux chiffres hexadécimaux dans la plage 00-FF, précédés de `x`. Plusieurs octets doivent être séparés par des virgules. Exemple :

```
srcrcdelimbytes="x08,xA4"
```

Vous pouvez spécifier l'attribut `srcrcdelimbytes` uniquement si le fichier source de transfert est orienté enregistrement, par exemple un ensemble de données z/OS , et si le fichier cible est un fichier normal, non orienté enregistrement. Vous ne pouvez pas spécifier l'attribut `srcrcdelimbytes` si vous avez également spécifié la valeur `text` pour l'attribut `conversion` .

srcrcdelimpos

Facultatif. Indique la position dans laquelle le délimiteur binaire est inséré. Les valeurs admises sont les suivantes :

- **prefix**-les délimiteurs sont insérés dans le fichier de destination avant les données de chaque enregistrement de fichier orienté enregistrement source.
- **postfix**-les délimiteurs sont insérés dans le fichier de destination après les données de chaque enregistrement de fichier orienté enregistrement source.

Vous pouvez spécifier l'attribut `srcrcdelimpos` uniquement si vous avez également spécifié l'attribut `srcrcdelimbytes` .

Attributs de l'option de destination

dstcodage

Facultatif. Codage de jeu de caractères à utiliser pour le fichier transféré.

Vous pouvez spécifier cet attribut uniquement lorsque l'attribut `conversion` est défini sur la valeur `text`.

Si l'attribut `dstencoding` n'est pas spécifié, le jeu de caractères du système de destination est utilisé pour les transferts de texte.

dsteol

Facultatif. Délimiteur de fin de ligne à utiliser pour le fichier transféré. Les valeurs admises sont les suivantes :

- `CRLF` -Utilisez un caractère de retour chariot suivi d'un caractère de saut de ligne comme délimiteur de fin de ligne. Cette convention est typique des systèmes Windows .
- `LF` -Utilisez un caractère de saut de ligne comme délimiteur de fin de ligne. Cette convention est typique des systèmes UNIX .

Vous pouvez spécifier cet attribut uniquement lorsque l'attribut `conversion` est défini sur la valeur `text`.

Si vous ne spécifiez pas l'attribut `dsteol` , les transferts de texte déterminent automatiquement la valeur correcte en fonction du système d'exploitation de l'agent de destination.

dstmsgdelimbytes

Facultatif. Indique le délimiteur hexadécimal à utiliser lors du fractionnement d'un fichier binaire en plusieurs messages. Tous les messages ont le même ID groupe IBM MQ ; le dernier message du groupe a l'indicateur IBM MQ `LAST_MSG_IN_GROUP` défini. Le format de spécification d'un octet hexadécimal comme délimiteur est `xNN`, où `N` est un caractère compris entre 0 et 9 ou a-f. Vous pouvez spécifier une séquence d'octets hexadécimaux comme délimiteur en spécifiant une liste d'octets hexadécimaux séparés par des virgules, par exemple: `x3e, x20, x20, xbf`.

Vous pouvez spécifier l'attribut `dstmsgdelimbytes` uniquement si vous avez également spécifié l'attribut `dstqueue` et que le transfert est en mode binaire. Vous ne pouvez spécifier qu'un seul des attributs `dstmsgsize`, `dstmsgdelimbyteset` et `dstmsgdelimpattern` .

dstmsgdelimpattern

Facultatif. Indique l'expression régulière Java à utiliser lors du fractionnement d'un fichier texte en plusieurs messages. Tous les messages ont le même ID groupe IBM MQ ; le dernier message du groupe a l'indicateur IBM MQ `LAST_MSG_IN_GROUP` défini. Le format de spécification d'une expression régulière comme délimiteur est une expression régulière entre parenthèses, (*regular_expression*) ou entre guillemets, "*regular_expression*". Pour plus d'informations, voir la section «[Expressions régulières utilisées par MFT](#)», à la page 679.

Par défaut, la longueur de la chaîne à laquelle l'expression régulière peut correspondre est limitée par l'agent de destination à cinq caractères. Vous pouvez modifier ce comportement à l'aide de la propriété d'agent `maxDelimiterMatchLength` . Pour plus d'informations, voir «[Propriétés d'agent avancées](#)», à la page 565.

Vous pouvez spécifier l'attribut `dstmsgdelimpattern` uniquement si vous avez également spécifié l'attribut `dstqueue` et que le transfert est en mode texte. Vous ne pouvez spécifier qu'un seul des attributs `dstmsgsize`, `dstmsgdelimbyteset` et `dstmsgdelimpattern` .

dstmsgdelimposition

Facultatif. Indique la position dans laquelle le texte ou le délimiteur binaire doit se trouver. Les valeurs admises sont les suivantes :

- `prefix` -Les délimiteurs sont attendus au début de chaque ligne.
- `postfix` -Les délimiteurs sont attendus à la fin de chaque ligne.

Vous pouvez spécifier l'attribut `dstmsgdelimposition` uniquement si vous avez également spécifié l'attribut `dstmsgdelimpattern` .

dstmsgincludedelim

Facultatif. Indique si le délimiteur utilisé pour fractionner le fichier en plusieurs messages doit être inclus dans les messages. Si l'attribut `dstmsgincludedelim` est spécifié, le délimiteur est inclus à la fin du message qui contient les données de fichier précédant le délimiteur. Par défaut, le délimiteur n'est pas inclus dans les messages. Vous pouvez spécifier l'attribut `dstmsgincludedelim` uniquement si vous avez également spécifié l'un des attributs `dstmsgdelimpattern` et `dstmsgdelimbytes` .

dstmsgpersistent

Facultatif. Indique si les messages écrits dans la file d'attente de destination sont persistants. Les valeurs admises sont les suivantes :

- `true` -Ecrit les messages persistants dans la file d'attente de destination. Il s'agit de la valeur par défaut.
- `false` -Ecrit des messages non persistants dans la file d'attente de destination.
- `qdef` -La valeur de persistance est extraite de l'attribut `DefPersistence` de la file d'attente de destination.

Vous pouvez spécifier cet attribut uniquement lorsque l'attribut `dstqueue` est également spécifié.

dstmsgprops

Facultatif. Indique si les propriétés de message IBM MQ sont définies pour le premier message écrit dans la file d'attente de destination par le transfert. Les valeurs possibles sont les suivantes :

- `true` -Définit les propriétés de message sur le premier message créé par le transfert.
- `false` -Ne pas définir de propriétés de message sur le premier message créé par le transfert. Il s'agit de la valeur par défaut.

Pour plus d'informations, voir la section «Propriétés de message MQ définies par MFT sur les messages écrits dans les files d'attente de destination», à la page 698.

Vous pouvez spécifier cet attribut uniquement lorsque l'attribut `dstqueue` est également spécifié.

taille_msgdst

Facultatif. Indique si le fichier doit être fractionné en plusieurs messages de longueur fixe. Tous les messages ont le même ID groupe IBM MQ ; l'indicateur IBM MQ `LAST_MSG_IN_GROUP` est défini pour le dernier message du groupe. La taille des messages est spécifiée par la valeur de `dstmsgsize`. Le format de `dstmsgsize` est *longueurunités*, où *longueur* est un entier positif et *unités* est l'une des valeurs suivantes:

- B -Octets. La valeur minimale autorisée est deux fois la valeur maximale en octets par caractère de la page de codes des messages de destination.
- K -Kibioctets. Cela équivaut à 1024 octets.
- M -Mébioctets. Cela équivaut à 1024 kibioctets.

Si le fichier est transféré en mode texte et se trouve dans un jeu de caractères codés sur deux octets ou dans un jeu de caractères codés sur plusieurs octets, il est scindé en messages sur la limite de caractères la plus proche de la taille de message spécifiée.

Vous pouvez spécifier l'attribut `dstmsgsize` uniquement si vous avez également spécifié l'attribut `dstqueue` . Vous ne pouvez spécifier qu'un seul des attributs `dstmsgsize`, `dstmsgdelimbyteset` et `dstmsgdelimpattern` .

dstunsupportedcodepage

Facultatif. Indique l'action à effectuer si le gestionnaire de files d'attente de destination, tel que spécifié par l'attribut `dstqueue` , ne prend pas en charge la page de codes utilisée lors du transfert de données de fichier vers une file d'attente en tant que transfert de texte. Les valeurs admises pour cet attribut sont les suivantes:

- `binary` -poursuit le transfert mais n'applique pas la conversion de page de codes aux données en cours de transfert. La spécification de cette valeur revient à ne pas définir l'attribut de conversion sur `text`.
- `fail` -ne pas poursuivre l'opération de transfert. Le fichier est enregistré comme n'ayant pas pu être transféré. Il s'agit de l'option par défaut.

Vous ne pouvez spécifier l'attribut `dstunsupportedcodepage` que si vous avez également spécifié l'attribut `dstqueue` et la valeur `text` pour l'attribut `conversion`.

dsttruncaterecords

Facultatif. Indique que les enregistrements de destination dont la longueur est supérieure à l'attribut de fichier `LRECL` sont tronqués. Si la valeur est `true`, les enregistrements sont tronqués. Si la valeur est `false`, les enregistrements sont encapsulés. Par défaut, le paramètre a la valeur `false`. Ce paramètre est valide uniquement pour les transferts en mode texte où la destination est un fichier.

Autres attributs

total de contrôle

Facultatif. Détermine l'algorithme utilisé pour la somme de contrôle des fichiers transférés.

- `MD5` -utilisez l'algorithme de hachage MD5.
- `NONE` -n'utilisez pas d'algorithme de total de contrôle.

Si vous ne spécifiez pas l'attribut `checksum`, la valeur par défaut `MD5` est utilisée.

Conversion

Facultatif. Indique le type de conversion à appliquer au fichier lors de son transfert. Les valeurs possibles sont les suivantes :

- `binary` -n'applique aucune conversion.
- `text` -applique la conversion de page de codes entre les systèmes source et cible. Appliquez également la conversion des délimiteurs de ligne. Les attributs `srcencoding`, `dstencoding`, `srceol` et `dsteol` influencent la conversion appliquée.

Si vous ne spécifiez pas l'attribut `conversion`, la valeur par défaut `binary` est spécifiée.

Ecraser

Facultatif. Détermine si un fichier de destination existant  ou un fichier peut être écrasé par l'opération. Lorsque vous spécifiez la valeur `true`, les fichiers de destination  ou les fichiers existants sont écrasés. Lorsque vous spécifiez la valeur `false`, l'existence d'un fichier en double  ou d'un fichier à la destination entraîne l'échec de l'opération. Si l'attribut `overwrite` n'est pas spécifié, la valeur par défaut `false` est spécifiée.

recurse

Facultatif. Détermine si le transfert de fichiers est récursif dans les sous-répertoires. Lorsque vous spécifiez la valeur `true`, le transfert est récursif dans les sous-répertoires. Lorsque vous spécifiez la valeur `false`, le transfert n'est pas récursif dans les sous-répertoires. Si l'attribut `recurse` n'est pas spécifié, la valeur par défaut `false` est spécifiée.

Exemple

Cet exemple spécifie `fte:filespec` avec un fichier source `file1.bin` et un fichier de destination `file2.bin`.

```
<fte:filespec srcfilespec="/home/fteuser/file1.bin" dstfile="/home/fteuser/file2.bin"/>
```

Référence associée

«Utilisation d' Apache Ant avec MFT», à la page 258

Managed File Transfer fournit des tâches que vous pouvez utiliser pour intégrer la fonction de transfert de fichier dans l'outil Ant Apache .

«Tâches Ant fournies par MFT», à la page 873

Managed File Transfer fournit un certain nombre de tâches Ant que vous pouvez utiliser pour accéder aux fonctions de transfert de fichiers.

fte:metadata Ant nested element

Les métadonnées sont utilisées pour transmettre des informations supplémentaires définies par l'utilisateur avec une opération de transfert de fichiers.

Pour plus d'informations sur la façon dont Managed File Transfer utilise les métadonnées, voir «Métadonnées pour les exits utilisateur MFT», à la page 898 .

Imbriqué par:

- La tâche [fte: filecopy](#)
- Tâche [fte: filemove](#)
- La tâche [fte: call](#)

Paramètres spécifiés en tant qu'éléments imbriqués

fte: entrée

Vous devez spécifier au moins une entrée dans l'élément imbriqué `fte:metadata` . Vous pouvez choisir d'indiquer plusieurs entrées. Les entrées associent un nom de clé à une valeur. Les clés doivent être uniques dans un bloc de `fte:metadata`

Attributs d'entrée

name

Obligatoire. Nom de la clé appartenant à cette entrée. Ce nom doit être unique dans tous les paramètres **entry** imbriqués dans un élément `fte: metadata` .

valeur

Obligatoire. Valeur à affecter à cette entrée.

Exemple

Cet exemple illustre une définition `fte:metadata` qui contient deux entrées.

```
<fte:metadata>
  <fte:entry name="org.foo.partColor" value="red"/>
  <fte:entry name="org.foo.partSize" value="medium"/>
</fte:metadata>
```

Référence associée

«Utilisation d' Apache Ant avec MFT», à la page 258

Managed File Transfer fournit des tâches que vous pouvez utiliser pour intégrer la fonction de transfert de fichier dans l'outil Ant Apache .

«Tâches Ant fournies par MFT», à la page 873

Managed File Transfer fournit un certain nombre de tâches Ant que vous pouvez utiliser pour accéder aux fonctions de transfert de fichiers.

Éléments imbriqués d'appel de programme

Les programmes peuvent être démarrés à l'aide de l'un des cinq éléments imbriqués: `fte:presrc`, `fte:predst`, `fte:postdst`, `fte:postsrct` et `fte:command`. Ces éléments imbriqués demandent à un agent d'appeler un programme externe dans le cadre de son traitement. Avant de démarrer un programme, vous devez vous assurer que la commande se trouve dans un emplacement spécifié par la propriété `commandPath` dans le fichier `agent.properties` de l'agent qui exécute la commande.

Même si chaque élément d'appel de programme a un nom différent, ils partagent le même ensemble d'attributs et le même ensemble d'éléments imbriqués. Les programmes peuvent être démarrés par les tâches Ant **fte:filecopy**, **fte:filemove** et **fte:command**.

Vous ne pouvez pas appeler de programmes à partir d'un agent de pont Connect:Direct.

Tâches Ant pouvant appeler des programmes:

- La tâche `fte:filecopy` imbrique les paramètres d'appel de programme à l'aide des éléments imbriqués `fte:predst`, `fte:postdst`, `fte:presrcet` et `fte:postsrc`.
- La tâche `fte:filemove` imbrique les paramètres d'appel de programme à l'aide des éléments imbriqués `fte:predst`, `fte:postdst`, `fte:presrcet` et `fte:postsrc`.
- La tâche `fte:call` imbrique les paramètres d'appel de programme à l'aide de l'élément imbriqué `fte:command`.

Attribut

commande

Obligatoire. Nomme le programme à appeler. Pour que l'agent puisse exécuter une commande, celle-ci doit se trouver dans un emplacement spécifié par la propriété `commandPath` dans le fichier `agent.properties` de l'agent. Pour plus d'informations, voir la section «[Propriété `commandPath` MFT](#)», à la page 374. Toute information de chemin spécifiée dans l'attribut `command` est considérée comme relative à un emplacement spécifié par la propriété `commandPath`. Lorsque `type` est `executable`, un programme exécutable est attendu, sinon un script approprié au type d'appel est attendu.

Nombre de relances

Facultatif. Nombre de tentatives d'appel du programme si le programme ne renvoie pas de code retour de réussite. Le programme nommé par l'attribut `command` est appelé jusqu'à ce nombre de fois. La valeur affectée à cet attribut doit être non négative. Si vous ne spécifiez pas l'attribut `retrycount`, la valeur par défaut zéro est utilisée.

attente de nouvelle tentative

Facultatif. Délai d'attente, en secondes, avant une nouvelle tentative d'appel du programme. Si le programme nommé par l'attribut `command` ne renvoie pas de code retour de réussite et que l'attribut `retrycount` indique une valeur différente de zéro, ce paramètre détermine le délai d'attente entre les nouvelles tentatives. La valeur affectée à cet attribut doit être non négative. Si vous ne spécifiez pas l'attribut `retrywait`, la valeur par défaut zéro est utilisée.

Code de réussite

Facultatif. La valeur de cet attribut est utilisée pour déterminer à quel moment l'appel de programme s'exécute correctement. Le code retour du processus pour la commande est évalué à l'aide de cette expression. La valeur peut être composée d'une ou de plusieurs expressions combinées avec un caractère de barre verticale (|) pour signifier la valeur booléenne OR ou une perluète (&) pour indiquer la valeur booléenne AND. Chaque expression peut être l'un des types d'expression suivants:

- Nombre indiquant un test d'égalité entre le code retour du processus et le nombre.
- Nombre précédé du caractère ">" pour indiquer un test supérieur à entre le nombre et le code retour du processus.
- Nombre précédé du caractère "<" pour indiquer un test inférieur à entre le nombre et le code retour du processus.
- Un nombre précédé d'un "!" pour indiquer un test non égal à entre le numéro et le code retour du processus.

Par exemple, `>2&<7&!5|0|14` est interprété comme les codes retour suivants ayant abouti: 0, 3, 4, 6, 14. Tous les autres codes retour sont interprétés comme n'ayant pas abouti. Si vous ne spécifiez pas l'attribut `successrc`, la valeur par défaut zéro est utilisée. Cela signifie que la commande est considérée comme ayant abouti si, et seulement si, elle renvoie un code de zéro.

type

Facultatif. La valeur de cet attribut indique le type de programme appelé. Indiquez l'une des options suivantes :

exécutable

La tâche appelle un programme exécutable. Peut avoir des arguments supplémentaires spécifiés à l'aide de l'élément imbriqué `arg`. Le programme doit être accessible sur le `commandPath` et, le cas échéant, disposer du droit d'exécution défini. Les scripts UNIX peuvent être appelés tant qu'ils spécifient un programme shell (par exemple, la première ligne du fichier script shell est: `#!/bin/sh`). La sortie de la commande écrite dans `stderr` ou `stdout` est envoyée au journal Managed File Transfer pour l'appel. Toutefois, la quantité de données générées est limitée par la configuration de l'agent. La valeur par défaut est 10K octets de données, mais vous pouvez remplacer cette valeur par défaut à l'aide de la propriété d'agent: `maxCommandOutput`.

antscript

La tâche exécute le script Ant spécifié à l'aide de la commande `fteAnt`. Les propriétés peuvent être spécifiées à l'aide de l'élément imbriqué `property`. Les cibles Ant peuvent être spécifiées à l'aide de l'élément imbriqué `target`. Le script Ant doit être accessible dans `commandPath`. La sortie Ant écrite dans `stderr` ou `stdout` est envoyée au journal Managed File Transfer pour l'appel. Toutefois, la quantité de données générées est limitée par la configuration de l'agent. La valeur par défaut est 10K octets de données, mais vous pouvez la remplacer à l'aide de la propriété d'agent: `maxCommandOutput`.

z/OS JCL

La valeur `jc1` est prise en charge sous z/OS uniquement et exécute le script JCL z/OS spécifié. Le JCL est soumis en tant que travail et requiert la présence de la carte de travail. Lorsque le travail est soumis avec succès, la sortie de la commande JCL, écrite dans le journal Managed File Transfer, contient le texte suivant: `JOB nom_travail(ID_travail)` où:

- `nom_travail` est le nom du travail identifié par la carte de travail dans le JCL.
- `job_id` est l'ID du travail généré par le système z/OS.

Si le travail ne peut pas être soumis correctement, la commande de script JCL échoue et écrit un message dans le journal indiquant la raison de l'échec (par exemple, aucune carte de travail n'est présente). Pour savoir si le travail a été exécuté ou exécuté avec succès, utilisez un service système tel que SDSF. Managed File Transfer ne fournit pas les informations car il soumet uniquement le travail ; le système détermine ensuite quand exécuter le travail et comment la sortie du travail est présentée. Etant donné qu'un script JCL est soumis en tant que travail par lots, il n'est pas conseillé de spécifier `jc1` pour un élément imbriqué `presrc` ou `predst` car vous savez uniquement que le travail a été soumis avec succès et non pas s'il a été exécuté avec succès avant le début du transfert. Aucun élément imbriqué n'est valide avec le type `jc1`.

L'exemple suivant illustre un travail JCL:

```
//MYJOB JOB
//*
//MYJOB EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=H
//SYSUT1 DD DSN=FRED.DEMO.TXT,DISP=SHR
//SYSUT2 DD DSN=BOB.DEMO.TXT,DISP=(NEW,CATLG),
// RECFM=VB,LRECL=133,BLKSIZE=2048,
// SPACE=(TRK,(30,5),RLSE)
//SYSIN DD DUMMY
```

Paramètres spécifiés en tant qu'éléments imbriqués

fte:arg

Valide uniquement lorsque la valeur de l'attribut `type` est `executable`. Utilisez des éléments `fte:arg` imbriqués pour spécifier des arguments pour le programme appelé dans le cadre de l'appel de programme. Les arguments de programme sont générés à partir des valeurs spécifiées par les éléments `fte:arg` dans l'ordre dans lequel les éléments `fte:arg` sont détectés. Vous pouvez

choisir de spécifier zéro ou plusieurs éléments `fte:arg` en tant qu'éléments imbriqués d'un appel de programme.

fte: propriété

Valide uniquement lorsque la valeur de l'attribut `type` est `antscript`. Utilisez les attributs `name` et `value` des éléments `fte:property` imbriqués pour transmettre des paires nom-valeur au script Ant. Vous pouvez choisir de spécifier zéro ou plusieurs éléments `fte:property` en tant qu'éléments imbriqués d'un appel de programme.

fte: cible

Valide uniquement lorsque la valeur de l'attribut `type` est `antscript`. Indiquez une cible dans le script Ant à appeler. Vous pouvez choisir de spécifier zéro ou plusieurs éléments `fte:target` en tant qu'éléments imbriqués d'un appel de programme.

Attributs d'argument

valeur

Obligatoire. Valeur de l'argument à transmettre au programme appelé.

Attributs de propriété

nom

Obligatoire. Nom d'une propriété à transmettre au script Ant.

valeur

Obligatoire. Valeur à associer au nom de propriété transmis au script Ant.

Exemples

Cet exemple illustre un appel de programme `fte:postsrc` spécifié dans le cadre d'une tâche `fte:filecopy`. L'appel de programme est destiné à un programme appelé `post.sh` et est fourni avec un seul argument de `/home/fteuser2/file.bin`.

```
<fte:filecopy cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  src="agent1@qm1" dst="agent2@qm2"
  rcproperty="copy.result">
  <fte:filespec srcfile="/home/fteuser1/file.bin" dstfile="/home/fteuser2/file.bin"/>

  <fte:postsrc command="post.sh" successsrc="1" >
    <fte:arg value="/home/fteuser2/file.bin"/>
  </fte:postsrc>
</fte:filecopy>
```

Cet exemple illustre un appel de programme `fte:command` spécifié dans le cadre d'une tâche `fte:call`. L'appel de programme est destiné à un exécutable appelé `command.sh`, qui ne transmet aucun argument de ligne de commande. Si `command.sh` ne renvoie pas le code retour de réussite 1, la commande est relancée au bout de 30 secondes.

```
<fte:call cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  agent="agent1@qm1"
  rcproperty="call.rc"
  origuser="bob"
  jobname="{j}job.id">
  <fte:command command="command.sh" successsrc="1" retrycount="5" retrywait="30"/>
</fte:call>
```

Cet exemple illustre un appel de programme `fte:command` spécifié dans le cadre d'une tâche `fte:call`. L'appel de programme concerne les cibles de copie et de compression dans un script Ant appelé `script.xml`, auquel deux propriétés sont transmises.

```
<fte:call cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  agent="agent1@qm1"
  rcproperty="call.rc"
  origuser="bob"
```

```

        jobname="${job.id}">
    <fte:command command="script.xml" type="antscript">
      <property name="src" value="AGENT5@QM5"/>
      <property name="dst" value="AGENT3@QM3"/>
      <target name="copy"/>
      <target name="compress"/>
    </fte:command>
  </fte:call>

```

Concepts associés

«Spécification des programmes à exécuter avec MFT», à la page 256

Vous pouvez exécuter des programmes sur un système sur lequel un Managed File Transfer Agent est en cours d'exécution. Dans le cadre d'une demande de transfert de fichiers, vous pouvez spécifier un programme à exécuter avant ou après le début d'un transfert. En outre, vous pouvez démarrer un programme qui ne fait pas partie d'une demande de transfert de fichier en soumettant une demande d'appel géré.

Référence associée

«Utilisation d' Apache Ant avec MFT», à la page 258

Managed File Transfer fournit des tâches que vous pouvez utiliser pour intégrer la fonction de transfert de fichier dans l'outil Ant Apache .

«Tâches Ant fournies par MFT», à la page 873

Managed File Transfer fournit un certain nombre de tâches Ant que vous pouvez utiliser pour accéder aux fonctions de transfert de fichiers.

Utilisation des exits utilisateur MFT pour la personnalisation

Métadonnées pour les exits utilisateur MFT

Il existe trois types de métadonnées qui peuvent être fournis aux routines d'exit utilisateur pour Managed File Transfer: métadonnées d'environnement, de transfert et de fichier. Ces métadonnées sont présentées sous forme de mappes de paires clé-valeur Java .

Métadonnées d'environnement

Les métadonnées d'environnement sont transmises à toutes les routines d'exit utilisateur et décrivent l'environnement d'exécution de l'agent à partir duquel la routine d'exit utilisateur est appelée. Ces métadonnées sont en lecture seule et ne peuvent pas être mises à jour par une routine d'exit utilisateur.

<i>Tableau 71. Métadonnées d'environnement</i>	
Clé	Description
CLÉ_RÉPERTOIRE_CONFIGURATION_AGENT	Nom du répertoire contenant les informations de configuration de l'agent.
CLE_REPERTOIRE_DU PRODUIT	Nom du répertoire dans lequel le code d'agent a été installé.
CL_VERSION_AGENT_	Numéro de version de l'environnement d'exécution de l'agent qui appelle la routine d'exit.

Les noms de clé et les noms de valeur indiqués dans le tableau 1 sont des constantes qui sont définies dans l'interface EnvironmentMetaDataConstants .

Métadonnées de transfert

Les métadonnées de transfert sont transmises à toutes les routines d'exit utilisateur. Les métadonnées sont constituées de valeurs fournies par le système et de valeurs fournies par l'utilisateur. Si vous modifiez des valeurs fournies par le système, ces modifications sont ignorées. Les valeurs initiales fournies par l'utilisateur pour l'exit utilisateur de début de transfert source sont basées sur les valeurs que vous fournissez lorsque vous définissez le transfert. L'agent source peut modifier les valeurs fournies par l'utilisateur dans le cadre du traitement de l'exit utilisateur de démarrage de transfert source. Cet

exit utilisateur est appelé avant le début du transfert complet de fichiers. Ces modifications sont utilisées dans les appels ultérieurs à d'autres routines d'exit liées à ce transfert. Les métadonnées de transfert sont appliquées à un transfert complet.

Bien que tous les exits utilisateur puissent lire les valeurs des métadonnées de transfert, seul l'exit utilisateur de démarrage de transfert source peut modifier les métadonnées de transfert

Vous ne pouvez pas utiliser les métadonnées de transfert pour propager des informations entre différents transferts de fichiers.

Les métadonnées de transfert fournies par le système sont détaillées dans le tableau 2:

<i>Tableau 72. Métadonnées de transfert</i>	
Clé	Description
CLÉ_AGENT_DESTINATION	Nom de l'agent qui est la destination du transfert.
CL_NOM_TRAVAIL	Nom de travail associé à la demande de transfert
CL_UTILISATEUR_MQDM	Zone utilisateur MQDM du message utilisé pour soumettre la demande de transfert
CLÉ_HÔTE_ORIGINE	Nom d'hôte spécifié comme nom d'hôte d'origine dans la demande de transfert
CL_UTILISATEUR_ORIGINATING_	Nom d'utilisateur spécifié comme ID utilisateur d'origine dans la demande de transfert
CL_AGENT_SOURCE	Nom de l'agent qui est la source du transfert
CLÉ D'ID_TRANSACTION	Identificateur du transfert

Les noms de clé et les noms de valeur indiqués dans le tableau 2 sont des constantes définies dans l'interface TransferMetaDataConstants .

Métadonnées de fichier

Les métadonnées de fichier sont transmises à l'exit de démarrage de transfert source dans le cadre de la spécification de fichier. Il existe des métadonnées de fichier distinctes pour les fichiers source et cible.

Vous ne pouvez pas utiliser les métadonnées de fichier pour propager des informations entre différents transferts de fichiers.

<i>Tableau 73. Métadonnées de fichier</i>		
Clé	Valeurs admises	Description
Séparateurs CONVERT_LINE_SEPARATORS		Valeur clé utilisée pour les transferts de texte pour indiquer si les séquences de séparateurs de ligne CRLF (retour chariot-saut de ligne) ou LF (saut de ligne) dans les données source sont converties en séquence de séparateurs de ligne à la destination.
CL_DELIMITER_DE		Valeur de clé utilisée pour définir un délimiteur afin de séparer les données d'enregistrement lors du transfert de données orientées enregistrement vers des fichiers normaux. Egalement utilisé pour les transferts de message à fichier et de fichier à message.

Tableau 73. Métadonnées de fichier (suite)		
Clé	Valeurs admises	Description
CLE_DELIMITER_POSITION_KEY	VALEUR_PREFIXE_POSITION_DELIMITER_POSITION_POS TFIX_VALUE	A utiliser avec DELIMITER_KEY pour définir la position du délimiteur (préfixe ou suffixe).
CLÉ_TYPE_DÉLIMITEUR	VALEUR_BINAIRE_TYPE_DÉLIMITEUR VALEUR_TEXTE_TYPE_DÉLIMITEUR VALEUR_TAILLE_TYPE_DÉLIMITEUR	A utiliser avec DELIMITER_KEY pour définir le type de délimiteur.
CLÉ_EXIST_DESTINATION	DESTINATION_EXIST_KEY_ERROR_VALUE DESTINATION_EXIST_KEY_OVERWRITE_VALUE	Détermine le comportement du transfert de fichier si le fichier de destination existe.
CL_ALIAS_FILE		Valeur de clé utilisée pour définir un alias pour le fichier en cours de transfert.
FILE_CHECKSUM_METHOD_KEY	FILE_CHECKSUM_METHOD_NONE_VALUE FILE_CHECKSUM_METHOD_MD5_VALUE	Détermine la méthode de total de contrôle à utiliser lors du transfert du fichier.
CL_CONVERSION_FILE_	VALEUR_TEXTE_CONVERSION_FICHER VALEUR_CONVERSION_BINAIRE_FICHER	Détermine le type de conversion appliqué au contenu du fichier.
CLÉ_ENCODING_FICHER		Détermine le codage utilisé pour un fichier texte.
FILE_END_OF_LINE_KEY (clé de ligne)	FILE_END_OF_LINE_LF_VALUE FILE_END_OF_LINE_CRLF_VALUE	Détermine la séquence de caractères qui indique la fin d'une ligne: < LF > ou < CR > < LF >.
ALIAS_ESPACE_FICHER		Détermine l'alias d'un fichier dans l'espace fichier. Remarque : Ces métadonnées ne peuvent être utilisées que si FILE_TYPE_KEY est FILE_TYPE_FILE_SPACE_VALUE
NOM_ESPACE_FICHER		Détermine le nom de l'espace fichier. Remarque : Ces métadonnées ne peuvent être utilisées que si FILE_TYPE_KEY est FILE_TYPE_FILE_SPACE_VALUE
CLE_TYPE_FICHER	FILE_TYPE_FILE_VALUE FILE_TYPE_DIRECTORY_VALUE FILE_TYPE_DATASET_VALUE FILE_TYPE_PDS_VALUE FILE_TYPE_QUEUE_VALUE FILE_TYPE_FILE_SPACE_VALUE	Détermine la spécification du fichier, de la file d'attente ou de l'espace fichier cible.
CL_ID_GROUPE		Valeur de clé utilisée pour les transferts de message à fichier afin de déterminer le groupe de messages à lire à partir de la file d'attente source. Cet attribut est valide uniquement lorsque la valeur de USE_GROUPS_KEY est USE_GROUPS_TRUE_VALUE.
INCLUDE_DELIMITER_IN_MESSAGE_KEY	INCLUDE_DELIMITER_IN_MESSAGE_TRUE_VALUE INCLUDE_DELIMITER_IN_MESSAGE_FALSE_VALUE	Valeur de clé utilisée pour les transferts de fichier à message afin de déterminer s'il convient d'inclure les délimiteurs utilisés pour fractionner le fichier en plusieurs messages à la fin des messages. Cet attribut est valide uniquement lorsque la valeur de DELIMITER_TYPE_KEY est DELIMITER_TYPE_BINARY_VALUE DELIMITER_TYPE_TEXT_VALUE.

Tableau 73. Métadonnées de fichier (suite)

Clé	Valeurs admises	Description
INSERT_RECORD_LINE_SEPARATOR_KEY		Valeur de clé utilisée pour les transferts de texte à partir de fichiers orientés enregistrement afin d'indiquer si des séparateurs de ligne sont insérés dans les données après chaque enregistrement.
CLÉ_ESPACE_DE_DÉBUT	KEEP_TRAILING_SPACES_TRUE_VALUE KEEP_TRAILING_SPACES_FALSE_VALUE	Valeur de clé utilisée pour déterminer si les espaces de fin sont supprimés des enregistrements lus à partir de fichiers de longueur fixe.
NOUVEAU_ENREGISTREMENT_SUR_LIGNE_CLÉ_SÉPARATEUR_CLÉ		Valeur clé utilisée pour les transferts de texte vers des fichiers orientés enregistrement afin d'indiquer si les séparateurs de ligne dans les données sont inclus avec les données d'enregistrement ou provoquent un nouvel enregistrement (et ne sont pas écrits).
CLE DE PERSISTANT	VALEUR_FICHER_PERSISTANT VALEUR DE PERSISTANT VALEUR PAR DEFAUT DE LA FILE D'ATTENTE	Valeur de clé utilisée pour les transferts de fichier à message afin de déterminer si les messages sont persistants.
CL_PROPS_MQ_SET_	VALEUR DE TRUE_SET_MQ_PROPS_ SET_MQ_PROPS_FALSE_VALUE	Valeur de clé utilisée pour les transferts de fichier à message afin de déterminer si les propriétés de message IBM MQ sont définies sur le premier message d'un fichier et sur tous les messages écrits dans la file d'attente en cas d'erreur.
CLÉ_PAGE_CODE_NON reconnaissable	VALEUR_ÉCHEC_PAGE_NON_RECONNAISSANCE VALEUR_ÉCHEC_CODE_PAGE_BINNAIRE_NON RECONNAISSABLE	Valeur de clé utilisée pour les transferts de fichier à message afin de déterminer si un transfert en mode texte échoue ou si la conversion est effectuée, si la page de codes des données n'est pas reconnue par le gestionnaire de files d'attente de destination.
CL_GROUPE_USE	USE_GROUPS_TRUE_VALUE VALEUR_UTILISATION_GROUPE_FALSE_VALUE	Valeur de clé utilisée pour les transferts de message à fichier afin de déterminer s'il convient de ne transférer qu'un groupe complet de messages de la file d'attente source.
CLE WAIT_TIME_CLE		Valeur de clé utilisée pour les transferts de message à fichier afin de déterminer le délai d'attente, en secondes, de l'agent source pour l'un des cas suivants: <ul style="list-style-type: none"> • Message à afficher dans la file d'attente source, si la file d'attente est vide ou est devenue vide, si la valeur de USE_GROUPS_KEY est FALSE. • Groupe complet à afficher dans la file d'attente source, si la valeur de USE_GROUPS_KEY est TRUE.

Les noms de clé et les noms de valeur indiqués dans le tableau 3 sont des constantes définies dans l'interface FileMetaDataConstants .

Exits utilisateur du moniteur de ressources MFT

Les exits utilisateur du moniteur de ressources permettent de configurer le code personnalisé à exécuter lorsque la condition de déclenchement d'un moniteur est remplie, avant le démarrage de la tâche associée.

Il n'est pas recommandé d'appeler de nouveaux transferts directement à partir du code d'exit utilisateur. Dans certaines circonstances, les fichiers sont transférés plusieurs fois car les exits utilisateur ne sont pas résilients aux redémarrages d'agent.

Les exits utilisateur du moniteur de ressources utilisent l'infrastructure existante pour les exits utilisateur. Les exits utilisateur du moniteur sont appelés après le déclenchement d'un moniteur, mais avant l'exécution de la tâche correspondante par la tâche du moniteur. Cela permet à l'exit utilisateur de modifier la tâche à exécuter et de décider si une tâche doit être exécutée ou non. Vous pouvez modifier la tâche de surveillance en mettant à jour les métadonnées de surveillance, qui sont ensuite utilisées pour la substitution de variables dans le document de tâche créé par la création du moniteur d'origine. L'exit du moniteur peut également remplacer ou mettre à jour la chaîne XML de définition de tâche transmise en tant que paramètre. L'exit du moniteur peut renvoyer le code de résultat 'continue' ou 'cancel' pour la tâche. Si l'annulation est renvoyée, la tâche ne sera pas démarrée et le moniteur ne redémarrera pas tant que la ressource contrôlée ne correspondra pas aux conditions de déclenchement. Si la ressource n'a pas été modifiée, le déclencheur ne démarre pas. Comme pour les autres exits utilisateur, vous pouvez chaîner les exits de surveillance. Si l'un des exits renvoie un code de résultat d'annulation, le résultat global est l'annulation et la tâche n'est pas démarrée.

- Une mappe de métadonnées d'environnement (identique à d'autres exits utilisateur)
- Mappe de métadonnées de surveillance incluant des métadonnées système non modifiables et des métadonnées utilisateur modifiables. Les métadonnées système non modifiables sont les suivantes:
 - FILENAME-Nom du fichier qui a satisfait à la condition de déclenchement
 - FILEPATH-chemin d'accès au fichier qui satisfait à la condition de déclenchement
 - FILESIZE (en octets-ces métadonnées peuvent ne pas être présentes)-taille du fichier qui satisfait à la condition de déclenchement
 - LASTMODIFIEDDATE (Local)-date à laquelle le fichier qui satisfait à la condition de déclenchement a été modifié pour la dernière fois. Cette date est exprimée par rapport à la date locale du fuseau horaire de l'agent et est formatée sous forme de date ISO 8601.
 - LASTMODIFIEDTIME (Local)-heure au format local à laquelle le fichier qui satisfait la condition de déclenchement a été modifié pour la dernière fois. Cette heure est exprimée par rapport à l'heure locale du fuseau horaire de l'agent et est formatée sous forme d'heure ISO 8601.
 - LASTMODIFIEDDATEUTC-date au format universel que le fichier qui satisfait à la condition de déclenchement a été modifié pour la dernière fois. Cette date est exprimée sous forme de date locale convertie au fuseau horaire TUC et est formatée sous forme de date ISO 8601.
 - LASTMODIFIEDTIMEUTC-heure au format universel à laquelle le fichier qui satisfait la condition de déclenchement a été modifié pour la dernière fois. Cette heure est exprimée sous forme d'heure locale convertie au fuseau horaire TUC et est formatée sous forme d'heure ISO 8601.
 - AGENTNAME-nom de l'agent de surveillance
- Chaîne XML représentant la tâche à exécuter suite au déclenchement du moniteur.

Les exits de surveillance renvoient les données suivantes:

- Un indicateur qui indique si la progression doit se poursuivre (continuer ou annuler)
- Chaîne à insérer dans le message de journal de déclenchement satisfait

Suite à l'exécution du code d'exit du moniteur, les métadonnées du moniteur et la chaîne XML de définition de tâche qui ont été initialement transmises en tant que paramètres peuvent également avoir été mises à jour.

La valeur de la propriété d'agent `monitorExitClasses` (dans le fichier `agent.properties`) indique les classes d'exit de surveillance à charger, chaque classe d'exit étant séparée par une virgule. Exemple :

```
monitorExitClasses=testExits.TestExit1,testExits.testExit2
```

L'interface de l'exit utilisateur du moniteur est la suivante:

```
package com.ibm.wmqfte.exitroutine.api;

import java.util.Map;

/**
 * An interface that is implemented by classes that want to be invoked as part of
 * user exit routine processing. This interface defines a method that will be
 * invoked immediately prior to starting a task as the result of a monitor trigger
 */
public interface MonitorExit {

    /**
     * Invoked immediately prior to starting a task as the result of a monitor
     * trigger.
     *
     * @param environmentMetaData
     *      meta data about the environment in which the implementation
     *      of this method is running. This information can only be read,
     *      it cannot be updated by the implementation. The constant
     *      defined in EnvironmentMetaDataConstants class can
     *      be used to access the data held by this map.
     *
     * @param monitorMetaData
     *      meta data to associate with the monitor. The meta data passed
     *      to this method can be altered, and the changes will be
     *      reflected in subsequent exit routine invocations. This map
     *      also contains keys with IBM reserved names. These entries are
     *      defined in the MonitorMetaDataConstants class and
     *      have special semantics. The the values of the IBM reserved names
     *      cannot be modified by the exit
     *
     * @param taskDetails
     *      An XML String representing the task to be executed as a result of
     *      the monitor triggering. This XML string may be modified by the
     *      exit
     *
     * @return
     *      a monitor exit result object which is used to determine if the
     *      task should proceed, or be cancelled.
     */
    MonitorExitResult onMonitor(Map<String, String> environmentMetaData,
                               Map<String, String> monitorMetaData,
                               Reference<String> taskDetails);
}

```

Les constantes des valeurs réservées à IBM dans les métadonnées du moniteur sont les suivantes:

```
package com.ibm.wmqfte.exitroutine.api;

/**
 * Constants for IBM reserved values placed into the monitor meta data
 * maps used by the monitor exit routines.
 */
public interface MonitorMetaDataConstants {

    /**
     * The value associated with this key is the name of the trigger
     * file associated with the monitor. Any modification performed
     * to this property by user exit routines will be ignored.
     */
}

```

```

*/
final String FILE_NAME_KEY = "FILENAME";

/**
 * The value associated with this key is the path to the trigger
 * file associated with the monitor. Any modification performed
 * to this property by user exit routines will be ignored.
 */
final String FILE_PATH_KEY = "FILEPATH";

/**
 * The value associated with this key is the size of the trigger
 * file associated with the monitor. This will not be present in
 * the cases where the size cannot be determined. Any modification
 * performed to this property by user exit routines will be ignored.
 */
final String FILE_SIZE_KEY = "FILESIZE";

/**
 * The value associated with this key is the local date on which
 * the trigger file associated with the monitor was last modified.
 * Any modification performed to this property by user exit routines
 * will be ignored.
 */
final String LAST_MODIFIED_DATE_KEY = "LASTMODIFIEDDATE";

/**
 * The value associated with this key is the local time at which
 * the trigger file associated with the monitor was last modified.
 * Any modification performed to this property by user exit routines
 * will be ignored.
 */
final String LAST_MODIFIED_TIME_KEY = "LASTMODIFIEDTIME";

/**
 * The value associated with this key is the UTC date on which
 * the trigger file associated with the monitor was last modified.
 * Any modification performed to this property by user exit routines
 * will be ignored.
 */
final String LAST_MODIFIED_DATE_KEY_UTC = "LASTMODIFIEDDATEUTC";

/**
 * The value associated with this key is the UTC time at which
 * the trigger file associated with the monitor was last modified.
 * Any modification performed to this property by user exit routines
 * will be ignored.
 */
final String LAST_MODIFIED_TIME_KEY_UTC = "LASTMODIFIEDTIMEUTC";

/**
 * The value associated with this key is the name of the agent on which
 * the monitor is running. Any modification performed to this property by
 * user exit routines will be ignored.
 */
final String MONITOR_AGENT_KEY = "AGENTNAME";
}

```

Exemple d'exit utilisateur de moniteur

Cet exemple de classe implémente l'interface `MonitorExit`. Cet exemple ajoute une variable de substitution personnalisée dans les métadonnées du moniteur appelées `REDIRECTEDAGENT` qui sera renseignée avec la valeur `LONDON` si l'heure de la journée est impaire et la valeur `PARIS` pour les heures paires. Le code de résultat de l'exit du moniteur est défini pour toujours renvoyer `proceed`.

```

package com.ibm.wmqfte.monitor;

import java.util.Calendar;
import java.util.Map;

import com.ibm.wmqfte.exitroutine.api.MonitorExit;
import com.ibm.wmqfte.exitroutine.api.MonitorExitResult;
import com.ibm.wmqfte.exitroutine.api.Reference;

/**
 * Example resource monitor user exit that changes the monitor mutable

```

```

* metadata value between 'LONDON' and 'PARIS' depending on the hour of the day.
*
*/
public class TestMonitorExit implements MonitorExit {

    // custom variable that will substitute destination agent
    final static String REDIRECTED_AGENT = "REDIRECTEDAGENT";

    public MonitorExitResult onMonitor(
Map<String, String> environmentMetaData,
Map<String, String> monitorMetaData,
Reference<String> taskDetails) {

        // always succeed
        final MonitorExitResult result = MonitorExitResult.PROCEED_RESULT;

        final int hour = Calendar.getInstance().get(Calendar.HOUR_OF_DAY);

        if (hour%2 == 1) {
            monitorMetaData.put(REDIRECTED_AGENT, "LONDON");
        } else {
            monitorMetaData.put(REDIRECTED_AGENT, "PARIS");
        }

        return result;
    }
}

```

La tâche correspondante pour un moniteur qui utilise la variable de substitution *REDIRECTEDAGENT* peut se présenter comme suit:

```

<?xml version="1.0" encoding="UTF-8"?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT1"
      QMgr="QM1"/>
    <destinationAgent agent="{REDIRECTEDAGENT}"
      QMgr="QM2"/>
    <transferSet>
      <item mode="binary" checksumMethod="MD5">
        <source recursive="false" disposition="delete">
          <file>c:\sourcefiles\reports.doc</file>
        </source>
        <destination type="file" exist="overwrite">
          <file>c:\destinationfiles\reports.doc</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>

```

Avant le début de ce transfert, la valeur de l'attribut agent de l'élément <destinationAgent> est remplacée par LONDON ou PARIS.

Vous devez spécifier la variable de substitution dans la classe d'exit du moniteur et dans le fichier XML de définition de tâche en majuscules.

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

«Métadonnées pour les exits utilisateur MFT», à la page 898

Il existe trois types de métadonnées qui peuvent être fournis aux routines d'exit utilisateur pour Managed File Transfer: métadonnées d'environnement, de transfert et de fichier. Ces métadonnées sont présentées sous forme de mappes de paires clé-valeur Java .

«Interfaces Java pour les exits utilisateur MFT», à la page 908

Utilisez les rubriques de cette section pour obtenir des informations de référence sur les interfaces Java pour les routines d'exit utilisateur.

Référence associée

«Propriétés de l'agent MFT pour les exits utilisateur», à la page 906

Outre les propriétés standard du fichier `agent.properties`, il existe plusieurs propriétés avancées spécifiques aux routines d'exit utilisateur. Ces propriétés ne sont pas incluses par défaut. Par conséquent, si vous souhaitez les utiliser, vous devez éditer manuellement le fichier `agent.properties`. Si vous modifiez le fichier `agent.properties` alors que cet agent est en cours d'exécution, arrêtez et redémarrez l'agent pour récupérer les modifications.

Propriétés de l'agent MFT pour les exits utilisateur

Outre les propriétés standard du fichier `agent.properties`, il existe plusieurs propriétés avancées spécifiques aux routines d'exit utilisateur. Ces propriétés ne sont pas incluses par défaut. Par conséquent, si vous souhaitez les utiliser, vous devez éditer manuellement le fichier `agent.properties`. Si vous modifiez le fichier `agent.properties` alors que cet agent est en cours d'exécution, arrêtez et redémarrez l'agent pour récupérer les modifications.

Pour IBM WebSphere MQ 7.5 ou les versions ultérieures, il est possible d'utiliser des variables d'environnement dans certaines propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction des changements d'environnement, tels que l'utilisateur qui exécute le processus. Pour plus d'informations, voir la section «Utilisation des variables d'environnement dans les propriétés MFT», à la page 547.

Propriétés de la routine d'exit utilisateur

Les routines d'exit utilisateur sont appelées dans l'ordre indiqué dans le tableau suivant. Pour plus d'informations sur le fichier `agent.properties`, voir [Propriétés avancées de l'agent: routine d'exit utilisateur](#).

Tableau 74. Propriétés d'agent pour les exits utilisateur	
Nom de la propriété	Description
<code>sourceTransferEndExitClasses</code>	Spécifie une liste de classes implémentant une routine d'exit de fin de transfert source.
<code>sourceTransferStartExitClasses</code>	Spécifie une liste de classes implémentant une routine d'exit de début de transfert source.
<code>destinationTransferStartExitClasses</code>	Spécifie une liste de classes implémentant une routine d'exit utilisateur de début de transfert de destination.
<code>destinationTransferEndExitClasses</code>	Indique une liste de classes séparées par des virgules qui implémentent une routine d'exit utilisateur de transfert de destination.
<code>exitClassPath</code>	<p>Indique une liste de répertoires, délimités par des caractères, spécifique à la plateforme, qui servent de chemins d'accès aux classes des routines d'exit utilisateur.</p> <p>Le répertoire des exits de l'agent est recherché avant toutes les entrées de ce chemin d'accès aux classes.</p> <p>Si vous utilisez cette propriété sous Windows, utilisez une barre oblique (/) comme délimiteur de chemin, et non la barre oblique inversée (\). Par exemple:</p> <pre>exitClassPath=C:/mycomp/mqft/exits/encryptFileExit.jar; C:/mycomp/mqft/exits/fileFilter.jar.</pre> <p>Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.</p>

Tableau 74. Propriétés d'agent pour les exits utilisateur (suite)

Nom de la propriété	Description
exitNativeLibraryPath	Indique une liste de répertoires, délimités par des caractères, spécifique à la plateforme, qui servent de chemins d'accès aux bibliothèques natives des routines d'exit utilisateur. Pour IBM WebSphere MQ 7.5 et les versions ultérieures, la valeur de cette propriété peut contenir des variables d'environnement.
monitorExitClasses	Spécifie une liste de classes implémentant une routine d'exit de moniteur, séparées par des virgules. Pour plus d'informations, voir la section «Exits utilisateur du moniteur de ressources MFT», à la page 902.
protocolBridgeCredentialExitClasses	Spécifie une liste de classes implémentant une routine d'exit utilisateur des données d'identification du pont de protocole. Pour plus d'informations, voir la section «Mappage des données d'identification pour un serveur de fichiers à l'aide de classes d'exit», à la page 230.
protocolBridgePropertiesExitClasses	Spécifie une liste de classes implémentant une routine d'exit utilisateur des propriétés serveur du pont de protocole. Pour plus d'informations, voir la section «ProtocolBridgePropertiesExit2: recherche des propriétés du serveur de fichiers de protocole», à la page 224.
IOExitClasses	Spécifie une liste de classes implémentant une routine d'exit utilisateur d'E-S, séparées par des virgules. Ne répertorie que les classes qui implémentent l'interface IOExit et non celles qui implémentent d'autres interfaces d'exit utilisateur d'E-S, comme IOExitResourcePath et IOExitChannel. Pour plus d'informations, voir la section «Utilisation des exits utilisateur d'E-S de transfert MFT», à la page 267.

Ordre d'appel de l'exit

Les exits source et de destination sont appelés dans l'ordre suivant:

1. SourceTransferStartExit
2. DestinationTransferStartExit
3. DestinationTransferEndExit
4. SourceTransferEndExit

Enchaînement des exits source et de destination

Si vous spécifiez plusieurs exits, le premier exit de la liste est appelé en premier, suivi du deuxième, et ainsi de suite. Toutes les modifications apportées par le premier exit sont transmises en tant qu'entrée à l'exit qui est ensuite appelé, etc. Par exemple, s'il existe deux exits de démarrage de transfert source, toutes les modifications apportées aux métadonnées de transfert par le premier exit sont entrées dans le deuxième exit. Chaque exit renvoie son propre résultat. Si tous les exits d'un type donné renvoient CONTINUE comme code de résultat de transfert, le résultat global est CONTINUE. Si un ou plusieurs exits renvoient CANCEL_TRANSFER, le résultat global est CANCEL_TRANSFER. Tous les codes de résultat et les chaînes renvoyés par les exits sont consignés dans le journal de transfert.

Si le résultat global de l'exit de démarrage de transfert source est CONTINUE, le transfert se poursuit en utilisant les modifications apportées par les exits. Si le résultat global est CANCEL_TRANSFER, les exits de fin de transfert source sont appelés, puis le transfert est annulé. Le statut d'achèvement dans le journal de transfert est "annulé".

Si le résultat global des exits de démarrage de transfert de destination est CONTINUE, le transfert se poursuit en utilisant les modifications apportées par les exits. Si le résultat global est CANCEL_TRANSFER, les exits de fin de transfert de destination sont appelés, puis les exits de fin de transfert source sont appelés. Enfin, le transfert est annulé. Le statut d'achèvement dans le journal de transfert est "annulé".

Si un exit source ou de destination doit transmettre des informations aux exits suivants dans la chaîne ou dans l'ordre d'exécution, il doit être effectué en mettant à jour les métadonnées de transfert. L'utilisation

des métadonnées de transfert est spécifique à l'implémentation de l'exit. Par exemple, si un exit définit le résultat de retour sur CANCEL_TRANSFER et doit communiquer aux exits suivants que le transfert a été annulé, il doit définir une valeur de métadonnées de transfert d'une manière comprise par les autres exits.

Exemple

```
sourceTransferStartExitClasses=com.ibm.wmqfte.test.MFTTestSourceTransferStartExit
sourceTransferEndExitClasses=com.ibm.wmqfte.test.MFTTestSourceTransferEndExit
destinationTransferStartExitClasses=com.ibm.wmqfte.test.MFTTestDestinationTransferStartExit
destinationTransferEndExitClasses=com.ibm.wmqfte.test.MFTTestDestinationTransferEndExit
exitClassPath=C:/mycomp/mqft/exits/encryptFileExit.jar;C:/mycomp/mqft/exits/fileFilter.jar
```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

«Métadonnées pour les exits utilisateur MFT», à la page 898

Il existe trois types de métadonnées qui peuvent être fournis aux routines d'exit utilisateur pour Managed File Transfer: métadonnées d'environnement, de transfert et de fichier. Ces métadonnées sont présentées sous forme de mappes de paires clé-valeur Java .

«Interfaces Java pour les exits utilisateur MFT», à la page 908

Utilisez les rubriques de cette section pour obtenir des informations de référence sur les interfaces Java pour les routines d'exit utilisateur.

Référence associée

«Exits utilisateur du moniteur de ressources MFT», à la page 902

Les exits utilisateur du moniteur de ressources permettent de configurer le code personnalisé à exécuter lorsque la condition de déclenchement d'un moniteur est remplie, avant le démarrage de la tâche associée.

«Utilisation des variables d'environnement dans les propriétés MFT», à la page 547

Depuis la IBM WebSphere MQ 7.5, il est possible d'utiliser des variables d'environnement dans les propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction des changements d'environnement. Par exemple, l'utilisateur qui exécute le processus.

«Le fichier MFT agent.properties», à la page 564

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.

Interfaces Java pour les exits utilisateur MFT

Utilisez les rubriques de cette section pour obtenir des informations de référence sur les interfaces Java pour les routines d'exit utilisateur.

Interface CDCredentialExit.java

CDCredentialExit.java

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
```

```

* disclosure restricted by GSA ADP Schedule Contract with
* IBM Corp.
*/
package com.ibm.wmqfte.exitroutine.api;

import java.util.Map;

/**
 * An interface that is implemented by classes that are invoked as part of
 * user exit routine processing. This interface defines methods that are
 * invoked by a Connect:Direct bridge agent to map the IBM MQ user ID of the transfer to credentials
 * that are used to access the Connect:Direct node.
 * There will be one instance of each implementation class per Connect:Direct bridge agent. The methods
 * can be called from different threads so the methods must be synchronized.
 */
public interface CDCredentialExit {

    /**
     * Invoked once when a Connect:Direct bridge agent is started. It is intended to initialize
     * any resources that are required by the exit
     *
     * @param bridgeProperties
     *        The values of properties defined for the Connect:Direct bridge.
     *        These values can only be read, they cannot be updated by
     *        the implementation.
     *
     * @return true if the initialisation is successful and false if unsuccessful
     *         If false is returned from an exit the Connect:Direct bridge agent does not
     *         start.
     */
    public boolean initialize(final Map<String, String> bridgeProperties);

    /**
     * Invoked once per transfer to map the IBM MQ user ID in the transfer message to the
     * credentials to be used to access the Connect:Direct node.
     *
     * @param mqUserId The IBM MQ user ID from which to map to the credentials to be used
     *                 to access the Connect:Direct node
     * @param snode    The name of the Connect:Direct SNODE specified as the cdNode in the
     *                 file path. This is used to map the correct user ID and password for the
     *                 SNODE.
     * @return        A credential exit result object that contains the result of the map and
     *                 the credentials to use to access the Connect:Direct node
     */
    public CDCredentialExitResult mapMQUserId(final String mqUserId, final String snode);

    /**
     * Invoked once when a Connect:Direct bridge agent is shutdown. This method releases
     * any resources that were allocated by the exit
     *
     * @param bridgeProperties
     *        The values of properties defined for the Connect:Direct bridge.
     *        These values can only be read, they cannot be updated by
     *        the implementation.
     *
     * @return
     */
    public void shutdown(final Map<String, String> bridgeProperties); }

```

Interface CredentialExitResult.java

CredentialExitResult.java

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */

```

```

package com.ibm.wmqfte.exitroutine.api;

/**
 * The result of invoking a Credential mapMQUserId exit method. It is composed of a result
 * code, which determines whether the mapping of the user id was successful, and an optional
 * Credentials object if the mapping is successful.
 */
public class CredentialExitResult {

    private final CredentialExitResultCode resultCode;
    private final Credentials credentials;

    /**
     * Constructor. Creates a credential exit result object with a specified result
     * code and optionally credentials.
     *
     * @param resultCode
     *         The result code to associate with the exit result being created.
     *
     * @param credentials
     *         The credentials to associate with the exit result being created.
     *         A value of <code>null</code> can be specified to indicate no
     *         credentials. If the resultCode is USER_SUCCESSFULLY_MAPPED the
     *         credentials must be set to a non-null value,
     */
    public CredentialExitResult(CredentialExitResultCode resultCode, Credentials credentials) {
        this.resultCode = resultCode;
        this.credentials = credentials;
    }

    /**
     * Returns the result code associated with this credential exit result
     *
     * @return the result code associated with this exit result.
     */
    public CredentialExitResultCode getResultCode() {
        return resultCode;
    }

    /**
     * Returns the credentials associated with this credential exit result
     *
     * @return the explanation associated with this credential exit result.
     */
    public Credentials getCredentials() {
        return credentials;
    }
}

```

Référence associée

[«Interface SourceTransferStartExit.java»](#), à la page 938

[«Interface DestinationTransferStartExit.java»](#), à la page 912

[«Interface DestinationTransferEndExit.java»](#), à la page 910

[«Interface MonitorExit.java»](#), à la page 931

[«Interface ProtocolBridgeCredentialExit.java»](#), à la page 932

Information associée

[Personnalisation de MFT avec des exits utilisateur](#)

Interface DestinationTransferEndExit.java

DestinationTransferEndExit.java

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2023. All Rights Reserved.

```

```

*
* US Government Users Restricted Rights - Use, duplication or
* disclosure restricted by GSA ADP Schedule Contract with
* IBM Corp.
*/
package com.ibm.wmqfte.exitpoint.api;

/**
 * An interface that is implemented by classes that want to be invoked as part of
 * user exit routine processing. This interface defines a method that will be
 * invoked immediately after completing a transfer on the agent acting as the
 * destination of the transfer.
 */
public interface DestinationTransferEndExit {

    /**
     * Invoked immediately after the completion of a transfer on the agent acting as
     * the destination of the transfer.
     *
     * @param transferExitResult
     *        a result object reflecting whether or not the transfer completed
     *        successfully.
     *
     * @param sourceAgentName
     *        the name of the agent acting as the source of the transfer.
     *
     * @param destinationAgentName
     *        the name of the agent acting as the destination of the
     *        transfer. This is the name of the agent that the
     *        implementation of this method will be invoked from.
     *
     * @param environmentMetaData
     *        meta data about the environment in which the implementation
     *        of this method is running. This information can only be read,
     *        it cannot be updated by the implementation. The constants
     *        defined in EnvironmentMetaDataConstants class can
     *        be used to access the data held by this map.
     *
     * @param transferMetaData
     *        meta data to associate with the transfer. The information can
     *        only be read, it cannot be updated by the implementation. This
     *        map may also contain keys with IBM reserved names. These
     *        entries are defined in the TransferMetaDataConstants
     *        class and have special semantics.
     *
     * @param fileResults
     *        a list of file transfer result objects that describe the source
     *        file name, destination file name and result of each file transfer
     *        operation attempted.
     *
     * @return
     *        an optional description to enter into the log message describing
     *        transfer completion. A value of null can be used
     *        when no description is required.
     */
    String onDestinationTransferEnd(TransferExitResult transferExitResult,
        String sourceAgentName,
        String destinationAgentName,
        Map<String, String>environmentMetaData,
        Map<String, String>transferMetaData,
        List<FileTransferResult>fileResults);
}

```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Référence associée

[«Interface SourceTransferStartExit.java», à la page 938](#)

[«Interface SourceTransferEndExit.java», à la page 937](#)

[«Interface DestinationTransferStartExit.java», à la page 912](#)

[«Interface MonitorExit.java», à la page 931](#)

[«Interface ProtocolBridgeCredentialExit.java», à la page 932](#)

DestinationTransferStartExit.java

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitpoint.api;

/**
 * An interface that is implemented by classes that want to be invoked as part of
 * user exit routine processing. This interface defines a method that will be
 * invoked immediately prior to starting a transfer on the agent acting as the
 * destination of the transfer.
 */
public interface DestinationTransferStartExit {

    /**
     * Invoked immediately prior to starting a transfer on the agent acting as
     * the destination of the transfer.
     *
     * @param sourceAgentName
     *     the name of the agent acting as the source of the transfer.
     *
     * @param destinationAgentName
     *     the name of the agent acting as the destination of the
     *     transfer. This is the name of the agent that the
     *     implementation of this method will be invoked from.
     *
     * @param environmentMetaData
     *     meta data about the environment in which the implementation
     *     of this method is running. This information can only be read,
     *     it cannot be updated by the implementation. The constants
     *     defined in EnvironmentMetaDataConstants class can
     *     be used to access the data held by this map.
     *
     * @param transferMetaData
     *     meta data to associate with the transfer. The information can
     *     only be read, it cannot be updated by the implementation. This
     *     map may also contain keys with IBM reserved names. These
     *     entries are defined in the TransferMetaDataConstants
     *     class and have special semantics.
     *
     * @param fileSpecs
     *     a list of file specifications that govern the file data to
     *     transfer. The implementation of this method can modify the
     *     entries in this list and the changes will be reflected in the
     *     files transferred. However, new entries may not be added and
     *     existing entries may not be removed.
     *
     * @return
     *     a transfer exit result object which is used to determine if the
     *     transfer should proceed, or be cancelled.
     */
    TransferExitResult onDestinationTransferStart(String sourceAgentName,
        String destinationAgentName,
        Map<String, String> environmentMetaData,
        Map<String, String> transferMetaData,
        List<Reference<String>> fileSpecs);
}
```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Référence associée

«Interface SourceTransferStartExit.java», à la page 938

[«Interface SourceTransferEndExit.java», à la page 937](#)

[«Interface DestinationTransferEndExit.java», à la page 910](#)

[«Interface MonitorExit.java», à la page 931](#)

[«Interface ProtocolBridgeCredentialExit.java», à la page 932](#)

Interface FileTransferResult.java

FileTransferResult.java

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */

package com.ibm.wmqfte.exitroutine.api;

/**
 * Result information about a file transfer.
 */
public interface FileTransferResult {

    /** An enumeration for the <code>getCorrelatorType()</code> method. */
    public enum CorrelationInformationType {
        /** No correlation information is available for this result */
        NONE,
        /**
         * The correlation information relates to work done in
         * IBM Sterling File Gateway.
         */
        SFG
    }

    /**
     * Returns the source file specification, from which the file was transferred.
     *
     * @return the source file specification, from which the file was
     * transferred.
     */
    String getSourceFileSpecification();

    /**
     * Returns the destination file specification, to which the file was transferred.
     *
     * @return the destination file specification, to which the file was
     * transferred. A value of <code>null</code> may be returned
     * if the transfer did not complete successfully.
     */
    String getDestinationFileSpecification();

    /**
     * Returns the result of the file transfer operation.
     *
     * @return the result of the file transfer operation.
     */
    FileExitResult getExitResult();

    /**
     * @return an enumerated value that identifies the product to which this correlating
     * information relates.
     */
    CorrelationInformationType getCorrelatorType();

    /**
     * @return the first string component of the correlating identifier that relates
     * this transfer result to work done in another product. A value of null
     * may be returned either because the other product does not utilize a

```

```

    *      string based correlation information or because there is no correlation
    *      information.
    */
String getString1Correlator();

/**
 * @return the first long component of the correlating identifier that relates
 * this transfer result to work done in another product. A value of zero
 * is returned when there is no correlation information or the other
 * product does not utilize long based correlation information or because
 * the value really is zero!
 */
long getLong1Correlator();
}

```

Référence associée

«Interface [SourceTransferStartExit.java](#)», à la page 938

«Interface [DestinationTransferStartExit.java](#)», à la page 912

«Interface [DestinationTransferEndExit.java](#)», à la page 910

«Interface [MonitorExit.java](#)», à la page 931

«Interface [ProtocolBridgeCredentialExit.java](#)», à la page 932

Information associée

[Personnalisation de MFT avec des exits utilisateur](#)

Interface IOExit.java

IOExit.java

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.io.IOException;
import java.util.Map;

import com.ibm.wmqfte.exitroutine.api.IOExitRecordResourcePath.RecordFormat;

/**
 * An interface that is implemented by classes that you want to be invoked as
 * part of user exit routine processing. This interface defines methods that
 * will be invoked during transfers to perform the underlying file system I/O
 * work for WMQFTE transfers.
 * <p>
 * The {@link #initialize(Map)} method will be called once when the exit is
 * first installed. The WMQFTE agent properties are passed to this method, thus
 * enabling the exit to understand its environment.
 * <p>
 * The {@link #isSupported(String)} method will be invoked during WMQFTE
 * transfers to determine whether the user exit should be used. If the
 * {@link #isSupported(String)} method returns a value of {@code true}, the
 * {@link #newPath(String)} method will be invoked for the paths specified for
 * the transfer request. The returned {@link IOExitPath} instance from a
 * {@link #newPath(String)} method invocation will then be used by the WMQFTE
 * transfer to obtain information about the resource and to transfer data to or
 * from the resource.
 * <p>
 * To obtain transfer context for an I/O exit, a {@link SourceTransferStartExit}
 * or {@link DestinationTransferStartExit} as appropriate, should be installed
 * to enable information to be seen by this exit. The
 * {@link SourceTransferStartExit} or {@link DestinationTransferStartExit} are

```

```

* passed the transfer's environment, metadata, and a list of file
* specifications for the transfer. The paths for the file specifications are
* the paths passed to the I/O exit's {@link #newPath(String)} method.
* <p>
* Note also that the {@link #isSupported(String)} and {@link #newPath(String)}
* methods might be called at other times by a WMQFTE agent and not just during
* transfers. For example, at transfer setup time the I/O system is queried to
* resolve the full resource paths for transfer.
*/
public interface IOExit {

    /**
     * Invoked once when the I/O exit is first required for use. It is intended
     * to initialize any resources that are required by the exit.
     *
     * @param agentProperties
     *     The values of properties defined for the WMQFTE agent. These
     *     values can only be read, they cannot be updated by the
     *     implementation.
     * @return {@code true} if the initialization is successful and {@code
     *     false} if unsuccessful. If {@code false} is returned from an
     *     exit, the exit will not be used.
     */
    boolean initialize(final Map<String, String> agentProperties);

    /**
     * Indicates whether this I/O user exit supports the specified path.
     * <p>
     * This method is used by WMQFTE to determine whether the I/O user exit
     * should be used within a transfer. If no I/O user exit returns true for
     * this method, the default WMQFTE file I/O function will be used.
     *
     * @param path
     *     The path to the required I/O resource.
     * @return {@code true} if the specified path is supported by the I/O exit,
     *     {@code false} otherwise
     */
    boolean isSupported(String path);

    /**
     * Obtains a new {@link IOExitPath} instance for the specified I/O resource
     * path.
     * <p>
     * This method will be invoked by WMQFTE only if the
     * {@link #isSupported(String)} method has been called for the path and
     * returned {@code true}.
     *
     * @param path
     *     The path to the required I/O resource.
     * @return A {@link IOExitPath} instance for the specified path.
     * @throws IOException
     *     If the path cannot be created for any reason.
     */
    IOExitPath newPath(String path) throws IOException;

    /**
     * Obtains a new {@link IOExitPath} instance for the specified I/O resource
     * path and passes record format and length information required by the
     * WMQFTE transfer.
     * <p>
     * Typically this method will be called for the following cases:
     * <ul>
     * <li>A path where a call to {@link #newPath(String)} has previously
     * returned a {@link IOExitRecordResourcePath} instance and WMQFTE is
     * re-establishing a new {@link IOExitPath} instance for the path, from an
     * internally-serialized state. The passed recordFormat and recordLength
     * will be the same as those for the original
     * {@link IOExitRecordResourcePath} instance.</li>
     * <li>A transfer destination path where the source of the transfer is
     * record oriented. The passed recordFormat and recordLength will be the
     * same as those for the source.</li>
     * </ul>
     * The implementation can act on the record format and length information as
     * deemed appropriate. For example, for a destination agent if the
     * destination does not already exist and the source of the transfer is
     * record oriented, the passed recordFormat and recordLength information
     * could be used to create an appropriate record-oriented destination path.
     * If the destination path already exists, the passed recordFormat and
     * recordLength information could be used to perform a compatibility check
     * and throw an {@link IOException} if the path is not compatible. A
     * compatibility check could ensure that a record oriented path's record
     * format is the same as the passed record format or that the record length

```

```

* is greater or equal to the passed record length.
* <p>
* This method will be invoked by WMQFTE only if the
* {@link #isSupported(String)} method has been called for the path and
* returned {@code true}.
*
* @param path
*         The path to the required I/O resource.
* @param recordFormat
*         The advised record format.
* @param recordLength
*         The advised record length.
* @return A {@link IOExitPath} instance for the specified path.
* @throws IOException
*         If the path cannot be created for any reason. For example,
*         the passed record format or length is incompatible with the
*         path's actual record format or length.
*/
IOExitPath newPath(String path, RecordFormat recordFormat, int recordLength)
    throws IOException;

```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Référence associée

«Utilisation des exits utilisateur d'E-S de transfert MFT», à la page 267

Vous pouvez utiliser les exits utilisateur d'E-S de transfert Managed File Transfer pour configurer le code personnalisé afin d'effectuer le travail d'E-S de système de fichiers sous-jacent pour les transferts Managed File Transfer .

Interface IOExitChannel.java

IOExitChannel.java

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.io.IOException;
import java.nio.ByteBuffer;

/**
 * Represents a channel that enables data to be read from or written to an
 * {@link IOExitResourcePath} resource.
 */
public interface IOExitChannel {

    /**
     * Obtains the data size for the associated {@link IOExitResourcePath} in
     * bytes.
     *
     * @return The data size in bytes.
     * @throws IOException
     *         If a problem occurs while attempting obtain the size.
     */
    long size() throws IOException;

    /**
     * Closes the channel, flushing any buffered write data to the resource and
     * releasing any locks.
     */
}

```

```

* @throws RecoverableIOException
*         If a recoverable problem occurs while closing the resource.
*         This means that WMQFTE can attempt to recover the transfer.
* @throws IOException
*         If some other I/O problem occurs. For example, the channel might
*         already be closed.
*/
void close() throws RecoverableIOException, IOException;

/**
* Reads data from this channel into the given buffer, starting at this
* channel's current position, and updates the current position by the
* amount of data read.
* <p>
* Data is copied into the buffer starting at its current position and up to
* its limit. On return, the buffer's position is updated to reflect the
* number of bytes read.
*
* @param buffer
*         The buffer that the data is to be copied into.
* @return The number of bytes read, which might be zero, or -1 if the end of
*         data has been reached.
* @throws RecoverableIOException
*         If a recoverable problem occurs while reading the data. For a
*         WMQFTE transfer this means that it will attempt to recover.
* @throws IOException
*         If some other I/O problem occurs. For a WMQFTE transfer this
*         means that it will be failed.
*/
int read(ByteBuffer buffer) throws RecoverableIOException, IOException;

/**
* Writes data to this channel from the given buffer, starting at this
* channel's current position, and updates the current position by the
* amount of data written. The channel's resource is grown to accommodate
* the data, if necessary.
* <p>
* Data is copied from the buffer starting at its current position and up to
* its limit. On return, the buffer's position is updated to reflect the
* number of bytes written.
*
* @param buffer
*         The buffer containing the data to be written.
* @return The number of bytes written, which might be zero.
* @throws RecoverableIOException
*         If a recoverable problem occurs while writing the data. For a
*         WMQFTE transfer this means that it will attempt to recover.
* @throws IOException
*         If some other I/O problem occurs. For a WMQFTE transfer this
*         means that it will be failed.
*/
int write(ByteBuffer buffer) throws RecoverableIOException, IOException;

/**
* Forces any updates to this channel's resource to be written to its
* storage device.
* <p>
* This method is required to force changes to both the resource's content
* and any associated metadata to be written to storage.
*
* @throws RecoverableIOException
*         If a recoverable problem occurs while performing the force.
*         For a WMQFTE transfer this means that it will attempt to
*         recover.
* @throws IOException
*         If some other I/O problem occurs. For a WMQFTE transfer this
*         means that it will be failed.
*/
void force() throws RecoverableIOException, IOException;

/**
* Attempts to lock the entire resource associated with the channel for
* shared or exclusive access.
* <p>
* The intention is for this method not to block if the lock is currently
* unavailable.
*
* @param shared
*         {@code true} if a shared lock is required, {@code false} if an
*         exclusive lock is required.
* @return A {@link IOExitLock} instance representing the newly acquired
*         lock or null if the lock cannot be obtained.

```

```

    * @throws IOException
    *         If a problem occurs while attempting to acquire the lock.
    */
    IOExitLock tryLock(boolean shared) throws IOException;
}

```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Référence associée

«Utilisation des exits utilisateur d'E-S de transfert MFT», à la page 267

Vous pouvez utiliser les exits utilisateur d'E-S de transfert Managed File Transfer pour configurer le code personnalisé afin d'effectuer le travail d'E-S de système de fichiers sous-jacent pour les transferts Managed File Transfer .

Interface IOExitLock.java

IOExitLock.java

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.io.IOException;

/**
 * Represents a lock on a resource for either shared or exclusive access.
 * {@link IOExitLock} instances are returned from
 * {@link IOExitChannel#tryLock(boolean)} calls and WMQFTE will request the
 * release of the lock at the appropriate time during a transfer. Additionally, when
 * a {@link IOExitChannel#close()} method is called it will be the
 * responsibility of the channel to release any associated locks.
 */
public interface IOExitLock {

    /**
     * Releases the lock.
     * <p>
     * After this method has been successfully called the lock is to be deemed as invalid.
     *
     * @throws IOException
     *         If the channel associated with the lock is not open or
     *         another problem occurs while attempting to release the lock.
     */
    void release() throws IOException;

    /**
     * Indicates whether this lock is valid.
     * <p>
     * A lock is considered valid until its @ {@link #release()} method is
     * called or the associated {@link IOExitChannel} is closed.
     *
     * @return {@code true} if this lock is valid, {@code false} otherwise.
     */
    boolean isValid();

    /**
     * @return {@code true} if this lock is for shared access, {@code false} if
     *         this lock is for exclusive access.
     */
}

```

```
boolean isShared();  
}
```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Référence associée

«Utilisation des exits utilisateur d'E-S de transfert MFT», à la page 267

Vous pouvez utiliser les exits utilisateur d'E-S de transfert Managed File Transfer pour configurer le code personnalisé afin d'effectuer le travail d'E-S de système de fichiers sous-jacent pour les transferts Managed File Transfer .

Interface IOExitPath.java

IOExitPath.java

```
/*  
 * Licensed Materials - Property of IBM  
 *  
 * "Restricted Materials of IBM"  
 *  
 * 5724-H72  
 *  
 * © Copyright IBM Corp. 2011, 2023. All Rights Reserved.  
 *  
 * US Government Users Restricted Rights - Use, duplication or  
 * disclosure restricted by GSA ADP Schedule Contract with  
 * IBM Corp.  
 */  
package com.ibm.wmqfte.exitroutine.api;  
  
/**  
 * Represents an abstract path that can be inspected and queried by WMQFTE for  
 * transfer purposes.  
 * <p>  
 * There are two types of path supported:  
 * <ul>  
 * <li>{@link IOExitResourcePath} - Represents a path that denotes a data  
 * resource. For example, a file, directory, or group of database records.</li>  
 * <li>{@link IOExitWildcardPath} - Represents a wildcard path that can be  
 * expanded to multiple {@link IOExitResourcePath} instances.</li>  
 * </ul>  
 */  
public abstract interface IOExitPath {  
  
    /**  
     * Obtains the abstract path as a {@link String}.  
     *  
     * @return The abstract path as a {@link String}.  
     */  
    String getPath();  
  
    /**  
     * Obtains the name portion of this abstract path as a {@link String}.  
     * <p>  
     * For example, a UNIX-style file system implementation evaluates the  
     * path {@code /home/fteuser/file1.txt} as having a name of {@code  
     * file1.txt}.  
     *  
     * @return the name portion of this abstract path as a {@link String}.  
     */  
    String getName();  
  
    /**  
     * Obtains the parent path for this abstract path as a {@link String}.  
     * <p>  
     * For example, a UNIX-style file system implementation evaluates the  
     * path {@code /home/fteuser/file1.txt} as having a parent path of {@code  
     * /home/fteuser}.  
     *  
     * @return The parent portion of the path as a {@link String}.  
     */  
}
```

```

String getParent();

/**
 * Obtains the abstract paths that match this abstract path.
 * <p>
 * If this abstract path denotes a directory resource, a list of paths
 * for all resources within the directory are returned.
 * <p>
 * If this abstract path denotes a wildcard, a list of all paths
 * matching the wildcard are returned.
 * <p>
 * Otherwise null is returned, because this abstract path probably denotes a
 * single file resource.
 *
 * @return An array of {@link IOExitResourcePath}s that
 *         match this path, or null if this method is not applicable.
 */
IOExitResourcePath[] listPaths();
}

```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Référence associée

«Utilisation des exits utilisateur d'E-S de transfert MFT», à la page 267

Vous pouvez utiliser les exits utilisateur d'E-S de transfert Managed File Transfer pour configurer le code personnalisé afin d'effectuer le travail d'E-S de système de fichiers sous-jacent pour les transferts Managed File Transfer .

Interface IOExitProperties.java

IOExitProperties.java

```

/**
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

/**
 * Properties that determine how WMQFTE treats an {@link IOExitPath} for certain
 * aspects of I/O. For example, whether to use intermediate files.
 */
public class IOExitProperties {

    private boolean rereadSourceOnRestart = true;
    private boolean rechecksumSourceOnRestart = true;
    private boolean rechecksumDestinationOnRestart = true;
    private boolean useIntermediateFileAtDestination = true;
    private boolean requiresSingleThreadedChannelIO = false;

    /**
     * Determines whether the I/O exit implementation expects the resource to be
     * re-read from the start if a transfer is restarted.
     *
     * @return {@code true} if, on restart, the I/O exit expects the source
     *         resource to be opened at the beginning and re-read from the
     *         beginning (the {@link IOExitPath#openForRead(long)} method is
     *         always invoked with 0L as an argument). {@code false} if, on
     *         restart, the I/O exit expects the source to be opened at the
     *         offset that the source agent intends to start reading from (the
     *         {@link IOExitPath#openForRead(long)} method can be invoked with a
     *         non-zero value as its argument).
     */
}

```

```

*/
public boolean getRereadSourceOnRestart() {
    return rereadSourceOnRestart;
}

/**
 * Sets the value to determine whether the I/O exit implementation expects
 * the resource to be re-read from the beginning if a transfer is restarted.
 * <p>
 * The default is {@code true}. The I/O exit should call this method when
 * required to change this value.
 *
 * @param rereadSourceOnRestart
 *     {@code true} if, on restart, the I/O exit expects the source
 *     resource to be opened at the beginning and re-read from the
 *     beginning (the {@link IOExitPath#openForRead(long)} method
 *     is always invoked with 0L as an argument). {@code false}
 *     if, on restart, the I/O exit expects the source to be opened
 *     at the offset that the source agent intends to start reading
 *     from (the {@link IOExitPath#openForRead(long)} method can be
 *     invoked with a non-zero value as its argument).
 */
public void setRereadSourceOnRestart(boolean rereadSourceOnRestart) {
    this.rereadSourceOnRestart = rereadSourceOnRestart;
}

/**
 * Determines whether the I/O exit implementation requires the source
 * resource to be re-checksummed if the transfer is restarted.
 * Re-checksumming takes place only if the
 * {@link #getRereadSourceOnRestart()} method returns {@code true}.
 *
 * @return {@code true} if, on restart, the I/O exit expects the already-
 *         transferred portion of the source to be re-checksummed for
 *         inconsistencies. Use this option in environments
 *         where the source could be changed during a restart. {@code
 *         false} if, on restart, the I/O exit does not require the
 *         already-transferred portion of the source to be re-checksummed.
 */
public boolean getRechecksumSourceOnRestart() {
    return rechecksumSourceOnRestart;
}

/**
 * Sets the value to determine whether the I/O exit implementation requires
 * the source resource to be re-checksummed if the transfer is restarted.
 * Re-checksumming takes place only if the
 * {@link #getRereadSourceOnRestart()} method returns {@code true}.
 * <p>
 * The default is {@code true}. The I/O exit should call this method when
 * required to change this value.
 *
 * @param rechecksumSourceOnRestart
 *     {@code true} if, on restart, the I/O exit expects the already
 *     transferred portion of the source to be re-checksummed
 *     for inconsistencies. Use this option in environments
 *     where the source could be changed during a restart.
 *     {@code false} if, on restart, the I/O exit does not
 *     require the already-transferred portion of the source to be
 *     re-checksummed.
 */
public void setRechecksumSourceOnRestart(boolean rechecksumSourceOnRestart) {
    this.rechecksumSourceOnRestart = rechecksumSourceOnRestart;
}

/**
 * Determines whether the I/O exit implementation requires the destination
 * resource to be re-checksummed if the transfer is restarted.
 *
 * @return {@code true} if, on restart, the I/O exit expects the already
 *         transferred portion of the destination to be re-checksummed to
 *         check for inconsistencies. This option should be used in
 *         environments where the destination could have been changed while
 *         a restart is occurring. {@code false} if, on restart, the I/O exit
 *         does not require the already transferred portion of the
 *         destination to be re-checksummed.
 */
public boolean getRechecksumDestinationOnRestart() {
    return rechecksumDestinationOnRestart;
}

/**

```

```

* Sets the value to determine whether the I/O exit implementation requires
* the destination resource to be re-checksummed if the transfer is
* restarted.
* <p>
* The default is {@code true}. The I/O exit should call this method when
* required to change this value.
*
* @param rechecksumDestinationOnRestart
*     {@code true} if, on restart, the I/O exit expects the already-
*     transferred portion of the destination to be re-checksummed
*     for inconsistencies. Use this option in environments
*     where the destination could have been changed during a
*     restart. {@code false} if, on restart, the I/O exit does not
*     require the already-transferred portion of the destination
*     to be re-checksummed.
*/
public void setRechecksumDestinationOnRestart(
    boolean rechecksumDestinationOnRestart) {
    this.rechecksumDestinationOnRestart = rechecksumDestinationOnRestart;
}

/**
* Determines whether the I/O exit implementation requires the use of an
* intermediate file when writing the data at the destination. The
* intermediate file mechanism is typically used to prevent an incomplete
* destination resource from being processed.
*
* @return {@code true} if data should be written to an intermediate file at
*         the destination and then renamed (to the requested destination
*         path name as specified in the transfer request) after the transfer is
*         complete. {@code false} if data should be written directly to the
*         requested destination path name without the use of an
*         intermediate file.
*/
public boolean getUseIntermediateFileAtDestination() {
    return useIntermediateFileAtDestination;
}

/**
* Sets the value to determine whether the I/O exit implementation requires
* the use of an intermediate file when writing the data at the destination.
* The intermediate file mechanism is typically used to prevent an
* incomplete destination resource from being processed.
*
* <p>
* The default is {@code true}. The I/O exit should call this method when
* required to change this value.
*
* @param useIntermediateFileAtDestination
*     {@code true} if data should be written to an intermediate file
*     at the destination and then renamed (to the requested
*     destination path name as specified in the transfer request) after
*     the transfer is complete. {@code false} if data should be written
*     directly to the requested destination path name without the
*     use of an intermediate file
*/
public void setUseIntermediateFileAtDestination(
    boolean useIntermediateFileAtDestination) {
    this.useIntermediateFileAtDestination = useIntermediateFileAtDestination;
}

/**
* Determines whether the I/O exit implementation requires
* {@link IOExitChannel} instances to be accessed by a single thread only.
*
* @return {@code true} if {@link IOExitChannel} instances are to be
*         accessed by a single thread only.
*/
public boolean requiresSingleThreadedChannelIO() {
    return requiresSingleThreadedChannelIO;
}

/**
* Sets the value to determine whether the I/O exit implementation requires
* channel operations for a particular instance to be accessed by a
* single thread only.
*
* <p>
* For certain I/O implementations it is necessary that resource path
* operations such as open, read, write, and close are invoked only from a
* single execution {@link Thread}. When set {@code true}, WMQFTE ensures
* that the following are invoked on a single thread:
* <ul>

```

```

* <li>{@link IOExitResourcePath#openForRead(long)} method and all methods of
* the returned {@link IOExitChannel} instance.</li>
* <li>{@link IOExitResourcePath#openForWrite(boolean)} method and all
* methods of the returned {@link IOExitChannel} instance.</li>
* </ul>
* <p>
* This has a slight performance impact, hence enable single-threaded channel
* I/O only when absolutely necessary.
* <p>
* The default is {@code false}. The I/O exit should call this method when
* required to change this value.
*
* @param requiresSingleThreadedChannelIO
*       {@code true} if {@link IOExitChannel} instances are to be
*       accessed by a single thread only.
*/
public void setRequiresSingleThreadedChannelIO(boolean requiresSingleThreadedChannelIO) {
    this.requiresSingleThreadedChannelIO = requiresSingleThreadedChannelIO;
}
}

```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Référence associée

«Utilisation des exits utilisateur d'E-S de transfert MFT», à la page 267

Vous pouvez utiliser les exits utilisateur d'E-S de transfert Managed File Transfer pour configurer le code personnalisé afin d'effectuer le travail d'E-S de système de fichiers sous-jacent pour les transferts Managed File Transfer .

Interface IOExitRecordChannel.java

IOExitRecordChannel.java

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.io.IOException;
import java.nio.ByteBuffer;

/**
 * Represents a channel that enables records of data to be read from or written
 * to an {@link IOExitRecordResourcePath} resource.
 * <p>
 * This is an extension of the {@link IOExitChannel} interface such that the
 * {@link #read(java.nio.ByteBuffer)} and {@link #write(java.nio.ByteBuffer)}
 * methods are expected to deal in whole records of data only. That is, the
 * {@link java.nio.ByteBuffer} returned from the read method and passed to the
 * write method is assumed to contain one or more complete records.
 */
public interface IOExitRecordChannel extends IOExitChannel {

    /**
     * Reads records from this channel into the given buffer, starting at this
     * channel's current position, and updates the current position by the
     * amount of data read.
     * <p>
     * Record data is copied into the buffer starting at its current position
     * and up to its limit. On return, the buffer's position is updated to
     * reflect the number of bytes read.
     */

```

```

* <p>
* Only whole records are copied into the buffer.
* <p>
* For a fixed-record-format resource, this might be multiple records. The
* amount of data in the return buffer does not necessarily need to be a
* multiple of the record length, but the last record is still to be treated
* as a complete record and padded as required by the caller.
* <p>
* For a variable-format resource, this is a single whole record of a size
* corresponding to the amount of return data or multiple whole records with
* all except the last being treated as records of maximum size.
*
* @param buffer
*     The buffer that the record data is to be copied into.
* @return The number of bytes read, which might be zero, or -1 if the end of
*     data has been reached.
* @throws RecoverableIOException
*     If a recoverable problem occurs while reading the data. For a
*     WMQFTE transfer this means that it will attempt to recover.
* @throws IOException
*     If some other I/O problem occurs, for example, if the passed
*     buffer is insufficient to contain at least one complete
*     record). For a WMQFTE transfer this means that it will be
*     failed.
*/
int read(ByteBuffer buffer) throws RecoverableIOException, IOException;

/**
* Writes records to this channel from the given buffer, starting at this
* channel's current position, and updates the current position by the
* amount of data written. The channel's resource is grown to accommodate
* the data, if necessary.
* <p>
* Record data is copied from the buffer starting at its current position
* and up to its limit. On return, the buffer's position is updated to
* reflect the number of bytes written.
* <p>
* The buffer is expected to contain only whole records.
* <p>
* For a fixed-record-format resource, this might be multiple records and if
* there is insufficient data in the buffer for a complete record, the
* record is to be padded as required to complete the record.
* <p>
* For a variable-record format resource the buffer is normally expected to
* contain a single record of length corresponding to the amount of data
* within the buffer. However, if the amount of data within the buffer
* exceeds the maximum record length, the implementation can either:
* <ol>
* <li>throw an {@link IOException} indicating that it cannot handle the
* situation.</li>
* <li>Consume a record's worth of data from the buffer, leaving the remaining
* data within the buffer.</li>
* <li>Consume all the buffer data and just write what it can to the current
* record. This effectively truncates the data.</li>
* <li>Consume all the buffer data and write to multiple records.</li>
* </ol>
*
* @param buffer
*     The buffer containing the data to be written.
* @return The number of bytes written, which might be zero.
* @throws RecoverableIOException
*     If a recoverable problem occurs while writing the data. For a
*     WMQFTE transfer this means that it will attempt to recover.
* @throws IOException
*     If some other I/O problem occurs. For a WMQFTE transfer this
*     means that it will be failed.
*/
int write(ByteBuffer buffer) throws RecoverableIOException, IOException;
}

```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Référence associée

«Utilisation des exits utilisateur d'E-S de transfert MFT», à la page 267

Vous pouvez utiliser les exits utilisateur d'E-S de transfert Managed File Transfer pour configurer le code personnalisé afin d'effectuer le travail d'E-S de système de fichiers sous-jacent pour les transferts Managed File Transfer .

 *Interface IOExitRecordResourcePath.java*

IOExitRecordResourcePath.java

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.io.IOException;

/**
 * Represents a path that denotes a record-oriented data resource (for example,
 * a z/OS data set). It allows the data to be located, the record format to be
 * understood, and {@link IOExitRecordChannel} instances to be created for read
 * or write operations.
 */
public interface IOExitRecordResourcePath extends IOExitResourcePath {

    /**
     * Record formats for record-oriented resources.
     */
    public enum RecordFormat {
        FIXED, VARIABLE
    }

    /**
     * Obtains the record length for records that are maintained by the resource
     * denoted by this abstract path.
     * <p>
     * For a resource with fixed-length records, the data for each record read
     * and written is assumed to be this length.
     * <p>
     * For a resource with variable-length records, this is the maximum length
     * for a record's data.
     * <p>
     * This method should return a value greater than zero, otherwise it can
     * result in the failure of a WMQFTE transfer that involves this abstract
     * path.
     *
     * @return The record length, in bytes, for records maintained by the
     *         resource.
     */
    int getRecordLength();

    /**
     * Obtains record format, as a {@link RecordFormat} instance, for records
     * that are maintained by the resource denoted by this abstract path.
     *
     * @return A {@link RecordFormat} instance for the record format for records
     *         that are maintained by the resource denoted by this abstract
     *         path.
     */
    RecordFormat getRecordFormat();

    /**
     * Opens a {@link IOExitRecordChannel} instance for reading data from the
     * resource denoted by this abstract path. The current data byte position
     * for the resource is expected to be the passed position value, such that
     * when {@link IOExitRecordChannel#read(java.nio.ByteBuffer)} is called,
     * data starting from that position is read.
     * <p>
     * Note that the data byte read position will be on a record boundary.
     */
}
```

```

*
* @param position
*       The required data byte read position.
* @return A new {@link IOExitRecordChannel} instance allowing data to be
*         read from the resource denoted by this abstract path.
* @throws RecoverableIOException
*         If a recoverable problem occurs while attempting to open the
*         resource for reading. This means that WMQFTE can attempt to
*         recover the transfer.
* @throws IOException
*         If some other I/O problem occurs.
*/
IOExitRecordChannel openForRead(long position)
    throws RecoverableIOException, IOException;

/**
* Opens a {@link IOExitRecordChannel} instance for writing data to the
* resource denoted by this abstract path. Writing of data, using the
* {@link IOExitRecordChannel#write(java.nio.ByteBuffer)} method, starts at
* either the beginning of the resource or end of the current data for the
* resource, depending on the specified append parameter.
*
* @param append
*       When {@code true} indicates that data written to the resource
*       should be appended to the end of the current data. When
*       {@code false} indicates that writing of data is to start at
*       the beginning of the resource; any existing data is lost.
* @return A new {@link IOExitRecordChannel} instance allowing data to be
*         written to the resource denoted by this abstract path.
* @throws RecoverableIOException
*         If a recoverable problem occurs while attempting to open the
*         resource for writing. This means that WMQFTE can attempt to
*         recover the transfer.
* @throws IOException
*         If some other I/O problem occurs.
*/
IOExitRecordChannel openForWrite(boolean append)
    throws RecoverableIOException, IOException;
}

```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Référence associée

«Utilisation des exits utilisateur d'E-S de transfert MFT», à la page 267

Vous pouvez utiliser les exits utilisateur d'E-S de transfert Managed File Transfer pour configurer le code personnalisé afin d'effectuer le travail d'E-S de système de fichiers sous-jacent pour les transferts Managed File Transfer .

Interface IOExitResourcePath.java

IOExitResourcePath.java

```

/*
* Licensed Materials - Property of IBM
*
* "Restricted Materials of IBM"
*
* 5724-H72
*
* © Copyright IBM Corp. 2011, 2023. All Rights Reserved.
*
* US Government Users Restricted Rights - Use, duplication or
* disclosure restricted by GSA ADP Schedule Contract with
* IBM Corp.
*/
package com.ibm.wmqfte.exitroutine.api;

import java.io.IOException;

/**
* Represents a path that denotes a data resource (for example, a file,

```

```

* directory, or group of database records). It allows the data to be located
* and IOExitChannel instances to be created for read or write
* operations.
* <p>
* There are two types of data resources as follows:
* <ul>
* <li>Directory - a container for other data resources. The
* isDirectory\(\) method returns true for these.</li>
* <li>File - a data container. This allows data to be read from or written to
* it. The isFile\(\) method returns true for these.</li>
* </ul>
*/
public interface IOExitResourcePath extends IOExitPath {

    /**
     * Creates a new IOExitResourcePath instance for a child path of the
     * resource denoted by this abstract path.
     * <p>
     * For example, with a UNIX-style path, IOExitResourcePath("/home/fteuser/test").newPath("subtest") could be
     * equivalent to: IOExitResourcePath("/home/fteuser/test/subtest")
     *
     * @param child
     *         The child path name.
     * @return A new IOExitResourcePath instance that represents a child
     *         of this path.
     */
    IOExitResourcePath newPath(final String child);

    /**
     * Creates the directory path for the resource denoted by this abstract
     * path, including any necessary but nonexistent parent directories. If the
     * directory path already exists, this method has no effect.
     * <p>
     * If this operation fails, it might have succeeded in creating some of the
     * necessary parent directories.
     *
     * @throws IOException
     *         If the directory path cannot be fully created, when it does
     *         not already exist.
     */
    void makePath() throws IOException;

    /**
     * Obtains the canonical path of the abstract path as a String.
     * <p>
     * A canonical path is defined as being absolute and unique. For example,
     * the path can be represented as UNIX-style relative path: test/file.txt
     * but the absolute and unique canonical path representation
     * is: /home/fteuser/test/file.txt
     *
     * @return The canonical path as a String.
     * @throws IOException
     *         If the canonical path cannot be determined for any reason.
     */
    String getCanonicalPath() throws IOException;

    /**
     * Tests if this abstract path is an absolute path.
     * <p>
     * For example, a UNIX-style path, /home/fteuser/test is an absolute
     * path, whereas fteuser/test is not.
     *
     * @return true if this abstract path is an absolute path, false
     *         otherwise.
     */
    boolean isAbsolute();

    /**
     * Tests if the resource denoted by this abstract path exists.
     *
     * @return true if the resource denoted by this abstract path
     *         exists, false otherwise.
     * @throws IOException
     *         If the existence of the resource cannot be determined for any
     *         reason.
     */
    boolean exists() throws IOException;

    /**
     * Tests whether the calling application can read the resource denoted by
     * this abstract path.

```

```

*
* @return {@code true} if the resource for this path exists and can be
*         read, {@code false} otherwise.
* @throws IOException
*         If a problem occurs while attempting to determine if the
*         resource can be read.
*/
boolean canRead() throws IOException;

/**
 * Tests whether the calling application can modify the resource denoted by
 * this abstract path.
 *
 * @return {@code true} if the resource for this path exists and can be
 *         modified, {@code false} otherwise.
 * @throws IOException
 *         If a problem occurs while attempting to determine if the
 *         resource can be modified.
 */
boolean canWrite() throws IOException;

/**
 * Tests whether the specified user is permitted to read the resource
 * denoted by this abstract path.
 * <p>
 * When WMQFTE invokes this method, the user identifier is the MQMD user
 * identifier for the requesting transfer.
 *
 * @param userId
 *         User identifier to test for access.
 * @return {@code true} if the resource for this abstract path exists and is
 *         permitted to be read by the specified user, {@code false}
 *         otherwise.
 * @throws IOException
 *         If a problem occurs while attempting to determine if the user
 *         is permitted to read the resource.
 */
boolean readPermitted(String userId) throws IOException;

/**
 * Tests whether the specified user is permitted to modify the resource
 * denoted by this abstract path.
 * <p>
 * When WMQFTE invokes this method, the user identifier is the MQMD user
 * identifier for the requesting transfer.
 *
 * @param userId
 *         User identifier to test for access.
 * @return {@code true} if the resource for this abstract path exists and is
 *         permitted to be modified by the specified user, {@code false}
 *         otherwise.
 * @throws IOException
 *         If a problem occurs while attempting to determine if the user
 *         is permitted to modify the resource.
 */
boolean writePermitted(String userId) throws IOException;

/**
 * Tests if the resource denoted by this abstract path is a directory-type
 * resource.
 *
 * @return {@code true} if the resource denoted by this abstract path is a
 *         directory type resource, {@code false} otherwise.
 */
boolean isDirectory();

/**
 * Creates the resource denoted by this abstract path, if it does not
 * already exist.
 *
 * @return {@code true} if the resource does not exist and was successfully
 *         created, {@code false} if the resource already existed.
 * @throws RecoverableIOException
 *         If a recoverable problem occurs while attempting to create
 *         the resource. This means that WMQFTE can attempt to recover
 *         the transfer.
 * @throws IOException
 *         If some other I/O problem occurs.
 */
boolean createNewPath() throws RecoverableIOException, IOException;
/**

```

```

* Tests if the resource denoted by this abstract path is a file-type
* resource.
*
* @return {@code true} if the resource denoted by this abstract path is a
*         file type resource, {@code false} otherwise.
*/
boolean isFile();

/**
 * Obtains the last modified time for the resource denoted by this abstract
 * path.
 * <p>
 * This time is measured in milliseconds since the epoch (00:00:00 GMT,
 * January 1, 1970).
 *
 * @return The last modified time for the resource denoted by this abstract
 *         path, or a value of 0L if the resource does not exist or a
 *         problem occurs.
 */
long lastModified();

/**
 * Deletes the resource denoted by this abstract path.
 * <p>
 * If the resource is a directory, it must be empty for the delete to work.
 *
 * @throws IOException
 *         If the delete of the resource fails for any reason.
 */
void delete() throws IOException;

/**
 * Renames the resource denoted by this abstract path to the specified
 * destination abstract path.
 * <p>
 * The rename should still be successful if the resource for the specified
 * destination abstract path already exists and it is possible to replace
 * it.
 *
 * @param destination
 *        The new abstract path for the resource denoted by this
 *        abstract path.
 * @throws IOException
 *        If the rename of the resource fails for any reason.
 */
void renameTo(IOExitResourcePath destination) throws IOException;

/**
 * Creates a new path to use for writing to a temporary resource that did
 * not previously exist.
 * <p>
 * The implementation can choose the abstract path name for the temporary
 * resource. However, for clarity and problem diagnosis, the abstract path
 * name for the temporary resource should be based on this abstract path
 * name with the specified suffix appended and additional characters to make
 * the path unique (for example, sequence numbers), as required.
 * <p>
 * When WMQFTE transfers data to a destination it normally attempts to first
 * write to a temporary resource then on transfer completion renames the
 * temporary resource to the required destination. This method is called by
 * WMQFTE to create a new temporary resource path. The returned path should
 * be new and the resource should not previously exist.
 *
 * @param suffix
 *        Recommended suffix to use for the generated temporary path.
 *
 * @return A new {@link IOExitResourcePath} instance for the temporary
 *         resource path, that did not previously exist.
 * @throws RecoverableIOException
 *         If a recoverable problem occurs whilst attempting to create
 *         the temporary resource. This means that WMQFTE can attempt to
 *         recover the transfer.
 * @throws IOException
 *         If some other I/O problem occurs.
 */
IOExitResourcePath createTempPath(String suffix)
    throws RecoverableIOException, IOException;

/**
 * Opens a {@link IOExitChannel} instance for reading data from the resource
 * denoted by this abstract path. The current data byte position for the
 * resource is expected to be the passed position value, such that when

```

```

* {@link IOExitChannel#read(java.nio.ByteBuffer)} is called, data starting
* from that position is read.
*
* @param position
*     The required data byte read position.
* @return A new {@link IOExitChannel} instance allowing data to be read
*     from the resource denoted by this abstract path.
* @throws RecoverableIOException
*     If a recoverable problem occurs while attempting to open the
*     resource for reading. This means that WMQFTE can attempt to
*     recover the transfer.
* @throws IOException
*     If some other I/O problem occurs.
*/
IOExitChannel openForRead(long position) throws RecoverableIOException,
    IOException;

/**
* Opens a {@link IOExitChannel} instance for writing data to the resource
* denoted by this abstract path. Writing of data, using the
* {@link IOExitChannel#write(java.nio.ByteBuffer)} method, starts at either
* the beginning of the resource or end of the current data for the
* resource, depending on the specified append parameter.
*
* @param append
*     When {@code true} indicates that data written to the resource
*     should be appended to the end of the current data. When
*     {@code false} indicates that writing of data is to start at
*     the beginning of the resource; any existing data is lost.
* @return A new {@link IOExitChannel} instance allowing data to be written
*     to the resource denoted by this abstract path.
* @throws RecoverableIOException
*     If a recoverable problem occurs whilst attempting to open the
*     resource for writing. This means that WMQFTE can attempt to
*     recover the transfer.
* @throws IOException
*     If some other I/O problem occurs.
*/
IOExitChannel openForWrite(boolean append) throws RecoverableIOException,
    IOException;

/**
* Tests if the resource denoted by this abstract path is in use by another
* application. Typically, this is because another application has a lock on
* the resource either for shared or exclusive access.
*
* @return {@code true} if resource denoted by this abstract path is in use
*     by another application, {@code false} otherwise.
*/
boolean inUse();

/**
* Obtains a {@link IOExitProperties} instance for properties associated
* with the resource denoted by this abstract path.
* <p>
* WMQFTE will read these properties to govern how a transfer behaves when
* interacting with the resource.
*
* @return A {@link IOExitProperties} instance for properties associated
*     with the resource denoted by this abstract path.
*/
IOExitProperties getProperties();
}

```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Référence associée

«Utilisation des exits utilisateur d'E-S de transfert MFT», à la page 267

Vous pouvez utiliser les exits utilisateur d'E-S de transfert Managed File Transfer pour configurer le code personnalisé afin d'effectuer le travail d'E-S de système de fichiers sous-jacent pour les transferts Managed File Transfer .

Interface IOExitWildcardPath.java

IOExitWildcardPath.java

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * Copyright IBM Corp. 2011, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

/**
 * Represents a path that denotes a wildcard. This can be used to match multiple
 * resource paths.
 */
public interface IOExitWildcardPath extends IOExitPath {
```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Référence associée

«Utilisation des exits utilisateur d'E-S de transfert MFT», à la page 267

Vous pouvez utiliser les exits utilisateur d'E-S de transfert Managed File Transfer pour configurer le code personnalisé afin d'effectuer le travail d'E-S de système de fichiers sous-jacent pour les transferts Managed File Transfer .

Interface MonitorExit.java

MonitorExit.java

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * Copyright IBM Corp. 2009, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.util.Map;

/**
 * An interface that is implemented by classes that want to be invoked as part of
 * user exit routine processing. This interface defines a method that will be
 * invoked immediately prior to starting a task as the result of a monitor trigger
 */
public interface MonitorExit {

    /**
     * Invoked immediately prior to starting a task as the result of a monitor
     * trigger.
     *
     * @param environmentMetaData
     * meta data about the environment in which the implementation
     * of this method is running. This information can only be read,
     * it cannot be updated by the implementation. The constant
     * defined in <code>EnvironmentMetaDataConstants</code> class can
     * be used to access the data held by this map.
     */
}
```

```

    * @param monitorMetaData
    *      meta data to associate with the monitor. The meta data passed
    *      to this method can be altered, and the changes will be
    *      reflected in subsequent exit routine invocations. This map
    *      also contains keys with IBM reserved names. These entries are
    *      defined in the MonitorMetaDataConstants class and
    *      have special semantics. The the values of the IBM reserved names
    *      cannot be modified by the exit
    *
    * @param taskDetails
    *      An XML String representing the task to be executed as a result of
    *      the monitor triggering. This XML string may be modified by the
    *      exit
    *
    * @return
    *      a monitor exit result object which is used to determine if the
    *      task should proceed, or be cancelled.
    */
    MonitorExitResult onMonitor(Map<String, String> environmentMetaData,
                               Map<String, String> monitorMetaData,
                               Reference<String> taskDetails);
}

```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Tâches associées

«Surveillance des ressources MFT», à la page 174

Vous pouvez surveiller les ressources Managed File Transfer ; par exemple, une file d'attente ou un répertoire. Lorsqu'une condition est satisfaite sur cette ressource, le moniteur de ressources démarre une tâche, telle qu'un transfert de fichier. Vous pouvez créer un moniteur de ressources à l'aide de la commande **fteCreateMonitor** ou de la vue **Moniteurs** dans le plug-in Managed File Transfer pour IBM MQ Explorer.

Référence associée

«Interface SourceTransferStartExit.java», à la page 938

«Interface SourceTransferEndExit.java», à la page 937

«Interface DestinationTransferStartExit.java», à la page 912

«Interface DestinationTransferEndExit.java», à la page 910

«Interface ProtocolBridgeCredentialExit.java», à la page 932

Interface ProtocolBridgeCredentialExit.java

ProtocolBridgeCredentialExit.java

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.util.Map;

/**
 * An interface that is implemented by classes that are to be invoked as part of
 * user exit routine processing. This interface defines methods that will
 * be invoked by a protocol bridge agent to map the MQ user ID of the transfer to credentials
 * that are to be used to access the protocol server.
 * There will be one instance of each implementation class per protocol bridge agent. The methods

```

```

* can be called from different threads so the methods must be synchronized.
*/
public interface ProtocolBridgeCredentialExit {

    /**
     * Invoked once when a protocol bridge agent is started. It is intended to initialize
     * any resources that are required by the exit
     *
     * @param bridgeProperties
     *       The values of properties defined for the protocol bridge.
     *       These values can only be read, they cannot be updated by
     *       the implementation.
     *
     * @return true if the initialization is successful and false if unsuccessful
     *         If false is returned from an exit the protocol bridge agent will not
     *         start
     */

    public boolean initialize(final Map<String> bridgeProperties);

    /**
     * Invoked once for each transfer to map the MQ user ID in the transfer message to the
     * credentials to be used to access the protocol server
     *
     * @param mqUserId The MQ user ID from which to map to the credentials to be used
     *                 access the protocol server
     * @return A credential exit result object that contains the result of the map and
     *         the credentials to use to access the protocol server
     */

    public CredentialExitResult mapMQUserId(final String mqUserId);

    /**
     * Invoked once when a protocol bridge agent is shutdown. It is intended to release
     * any resources that were allocated by the exit
     *
     * @param bridgeProperties
     *       The values of properties defined for the protocol bridge.
     *       These values can only be read, they cannot be updated by
     *       the implementation.
     *
     * @return
     */

    public void shutdown(final Map<String> bridgeProperties);
}

```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Tâches associées

«Mappage des données d'identification pour un serveur de fichiers à l'aide de classes d'exit», à la page 230

Si vous ne souhaitez pas utiliser la fonction de mappage des données d'identification par défaut de l'agent de pont de protocole, vous pouvez mapper les données d'identification de l'utilisateur dans Managed File Transfer aux données d'identification de l'utilisateur sur le serveur de fichiers 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. Si vous configurez des exits utilisateur de mappage de données d'identification, ils prennent la place de la fonction de mappage de données d'identification par défaut.

Interface ProtocolBridgeCredentialExit2.java

ProtocolBridgeCredentialExit2.java

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 */

```

```

* 5724-H72
*
*   Copyright IBM Corp. 2011, 2023. All Rights Reserved.
*
*   US Government Users Restricted Rights - Use, duplication or
*   disclosure restricted by GSA ADP Schedule Contract with
*   IBM Corp.
*/
package com.ibm.wmqfte.exitroutine.api;

/**
 * An interface that is implemented by classes that are invoked as part of user
 * exit routine processing. This interface defines methods that are invoked by a
 * protocol bridge agent to map the MQ user ID of the transfer to credentials
 * used to access a specified protocol bridge server. There will be one instance
 * of each implementation class for each protocol bridge agent. The methods can
 * be called from different threads so the methods must be synchronized.
 */
public interface ProtocolBridgeCredentialExit2 extends
    ProtocolBridgeCredentialExit {

    /**
     * Invoked once for each transfer to map the MQ user ID in the transfer
     * message to the credentials used to access a specified protocol server.
     *
     * @param endPoint
     *         Information that describes the protocol server to be accessed.
     * @param mqUserId
     *         The MQ user ID from which to map the credentials used to
     *         access the protocol server.
     * @return A {@link CredentialExitResult} instance that contains the result
     *         of the map and the credentials to use to access the protocol
     *         server.
     */
    public CredentialExitResult mapMQUserId(
        final ProtocolServerEndPoint endPoint, final String mqUserId);
}

```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Tâches associées

«Mappage des données d'identification pour un serveur de fichiers à l'aide de classes d'exit», à la page 230

Si vous ne souhaitez pas utiliser la fonction de mappage des données d'identification par défaut de l'agent de pont de protocole, vous pouvez mapper les données d'identification de l'utilisateur dans Managed File Transfer aux données d'identification de l'utilisateur sur le serveur de fichiers 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. Si vous configurez des exits utilisateur de mappage de données d'identification, ils prennent la place de la fonction de mappage de données d'identification par défaut.

Interface ProtocolBridgePropertiesExit2.java

ProtocolBridgePropertiesExit2.java

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 *   Copyright IBM Corp. 2011, 2023. All Rights Reserved.
 *
 *   US Government Users Restricted Rights - Use, duplication or
 *   disclosure restricted by GSA ADP Schedule Contract with
 *   IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.util.Map;

```

```

import java.util.Properties;

/**
 * An interface that is implemented by classes that are to be invoked as part of
 * user exit routine processing. This interface defines methods that will be
 * invoked by a protocol bridge agent to look up properties for protocol servers
 * that are referenced in transfers.
 * <p>
 * There will be one instance of each implementation class for each protocol
 * bridge agent. The methods can be called from different threads so the methods
 * must be synchronised.
 */
public interface ProtocolBridgePropertiesExit2 {

    /**
     * Invoked once when a protocol bridge agent is started. It is intended to
     * initialize any resources that are required by the exit.
     *
     * @param bridgeProperties
     *     The values of properties defined for the protocol bridge.
     *     These values can only be read, they cannot be updated by the
     *     implementation.
     * @return {@code true} if the initialization is successful and {@code
     *     false} if unsuccessful. If {@code false} is returned from an exit
     *     the protocol bridge agent will not start.
     */
    public boolean initialize(final Map<String, String> bridgeProperties);

    /**
     * Invoked when the Protocol Bridge needs to access the protocol bridge credentials XML file.
     *
     * @return a {@link String} object giving the location of the ProtocolBridgeCredentials.xml
     */
    public String getCredentialLocation ();

    /**
     * Obtains a set of properties for the specified protocol server name.
     * <p>
     * The returned {@link Properties} must contain entries with key names
     * corresponding to the constants defined in
     * {@link ProtocolServerPropertyConstants} and in particular must include an
     * entry for all appropriate constants described as required.
     *
     * @param protocolServerName
     *     The name of the protocol server whose properties are to be
     *     returned. If a null or a blank value is specified, properties
     *     for the default protocol server are to be returned.
     * @return The {@link Properties} for the specified protocol server, or null
     *     if the server cannot be found.
     */
    public Properties getProtocolServerProperties(
        final String protocolServerName);

    /**
     * Invoked once when a protocol bridge agent is shut down. It is intended to
     * release any resources that were allocated by the exit.
     *
     * @param bridgeProperties
     *     The values of properties defined for the protocol bridge.
     *     These values can only be read, they cannot be updated by the
     *     implementation.
     */
    public void shutdown(final Map<String, String> bridgeProperties);
}

```

Concepts associés

[«ProtocolBridgePropertiesExit: recherche des propriétés du serveur de fichiers de protocole», à la page 226](#)

Si vous disposez d'un grand nombre de serveurs de fichiers de protocole, vous pouvez implémenter l'interface `com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit` pour rechercher les propriétés de serveur de fichiers de protocole référencées dans les transferts. Vous pouvez implémenter cette interface de préférence à la gestion d'un fichier `ProtocolBridgeProperties.xml`. Il est recommandé d'utiliser l'interface `ProtocolBridgePropertiesExit2.java`, mais l'interface `ProtocolBridgePropertiesExit.java` est également prise en charge. Si vous disposez d'une implémentation existante de l'interface `ProtocolBridgePropertiesExit.java` à partir de IBM WebSphere MQ File Transfer Edition, vous pouvez

l'utiliser dans IBM WebSphere MQ 7.5 ou version ultérieure. La nouvelle méthode d'emplacement `getCredential` dans `ProtocolBridgePropertiesExit2.java` utilise l'emplacement par défaut du fichier `ProtocolBridgeCredentials.xml`, qui correspond à votre répertoire de base.

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Tâches associées

«Mappage des données d'identification pour un serveur de fichiers à l'aide de classes d'exit», à la page 230

Si vous ne souhaitez pas utiliser la fonction de mappage des données d'identification par défaut de l'agent de pont de protocole, vous pouvez mapper les données d'identification de l'utilisateur dans Managed File Transfer aux données d'identification de l'utilisateur sur le serveur de fichiers 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. Si vous configurez des exits utilisateur de mappage de données d'identification, ils prennent la place de la fonction de mappage de données d'identification par défaut.

SourceFileExitFileSpecification.java

SourceFileExitFileSpecification.java

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2012, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.util.Map;

/**
 * A specification of the file names to use for a file transfer, as evaluated by the
 * agent acting as the source of the transfer.
 */
public final class SourceFileExitFileSpecification {

    private final String sourceFileSpecification;
    private final String destinationFileSpecification;
    private final Map<String, String> sourceFileMetaData;
    private final Map<String, String> destinationFileMetaData;

    /**
     * Constructor. Creates a source file exit file specification.
     *
     * @param sourceFileSpecification
     *         the source file specification to associate with the source file
     *         exit file specification.
     *
     * @param destinationFileSpecification
     *         the destination file specification to associate with the
     *         source file exit file specification.
     *
     * @param sourceFileMetaData
     *         the source file meta data.
     *
     * @param destinationFileMetaData
     *         the destination file meta data
     */
    public SourceFileExitFileSpecification(final String sourceFileSpecification,
                                          final String destinationFileSpecification,
                                          final Map<String, String> sourceFileMetaData,
                                          final Map<String, String> destinationFileMetaData) {
        this.sourceFileSpecification = sourceFileSpecification;
        this.destinationFileSpecification = destinationFileSpecification;
    }
}
```

```

        this.sourceFileMetaData = sourceFileMetaData;
        this.destinationFileMetaData = destinationFileMetaData;
    }

    /**
     * Returns the destination file specification.
     *
     * @return the destination file specification. This represents the location,
     *         on the agent acting as the destination for the transfer, where the
     *         file should be written. Exit routines installed into the agent
     *         acting as the destination for the transfer may override this value.
     */
    public String getDestination() {
        return destinationFileSpecification;
    }

    /**
     * Returns the source file specification.
     *
     * @return the source file specification. This represents the location where
     *         the file data will be read from.
     */
    public String getSource() {
        return sourceFileSpecification;
    }

    /**
     * Returns the file meta data that relates to the source file specification.
     *
     * @return the file meta data that relates to the source file specification.
     */
    public Map<String, String> getSourceFileMetaData() {
        return sourceFileMetaData;
    }

    /**
     * Returns the file meta data that relates to the destination file specification.
     *
     * @return the file meta data that relates to the destination file specification.
     */
    public Map<String, String> getDestinationFileMetaData() {
        return destinationFileMetaData;
    }
}

```

Concepts associés

«Métadonnées pour les exits utilisateur MFT», à la page 898

Il existe trois types de métadonnées qui peuvent être fournis aux routines d'exit utilisateur pour Managed File Transfer: métadonnées d'environnement, de transfert et de fichier. Ces métadonnées sont présentées sous forme de mappes de paires clé-valeur Java .

Interface SourceTransferEndExit.java

SourceTransferEndExit.java

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitpoint.api;

/**
 * An interface that is implemented by classes that want to be invoked as part of
 * user exit routine processing. This interface defines a method that will be
 * invoked immediately after completing a transfer on the agent acting as the
 * source of the transfer.
 */

```

```

public interface SourceTransferEndExit {

    /**
     * Invoked immediately after the completion of a transfer on the agent acting as
     * the source of the transfer.
     *
     * @param transferExitResult
     *        a result object reflecting whether or not the transfer completed
     *        successfully.
     *
     * @param sourceAgentName
     *        the name of the agent acting as the source of the transfer.
     *        This is the name of the agent that the implementation of this
     *        method will be invoked from.
     *
     * @param destinationAgentName
     *        the name of the agent acting as the destination of the
     *        transfer.
     *
     * @param environmentMetaData
     *        meta data about the environment in which the implementation
     *        of this method is running. This information can only be read,
     *        it cannot be updated by the implementation. The constants
     *        defined in EnvironmentMetaDataConstants class can
     *        be used to access the data held by this map.
     *
     * @param transferMetaData
     *        meta data to associate with the transfer. The information can
     *        only be read, it cannot be updated by the implementation. This
     *        map may also contain keys with IBM reserved names. These
     *        entries are defined in the TransferMetaDataConstants
     *        class and have special semantics.
     *
     * @param fileResults
     *        a list of file transfer result objects that describe the source
     *        file name, destination file name and result of each file transfer
     *        operation attempted.
     *
     * @return
     *        an optional description to enter into the log message describing
     *        transfer completion. A value of null can be used
     *        when no description is required.
     */
    String onSourceTransferEnd(TransferExitResult transferExitResult,
                               String sourceAgentName,
                               String destinationAgentName,
                               Map<String, String>environmentMetaData,
                               Map<String, String>transferMetaData,
                               List<FileTransferResult>fileResults);

}

```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Référence associée

[«Interface SourceTransferStartExit.java», à la page 938](#)

[«Interface DestinationTransferStartExit.java», à la page 912](#)

[«Interface DestinationTransferEndExit.java», à la page 910](#)

[«Interface MonitorExit.java», à la page 931](#)

[«Interface ProtocolBridgeCredentialExit.java», à la page 932](#)

Interface SourceTransferStartExit.java

SourceTransferStartExit.java

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"

```

```

*
* 5724-H72
*
*   Copyright IBM Corp. 2008, 2023. All Rights Reserved.
*
* US Government Users Restricted Rights - Use, duplication or
* disclosure restricted by GSA ADP Schedule Contract with
* IBM Corp.
*/
package com.ibm.wmqfte.exitpoint.api;

import java.util.List;
import java.util.Map;

/**
 * An interface that is implemented by classes that want to be invoked as part of
 * user exit routine processing. This interface defines a method that will be
 * invoked immediately prior to starting a transfer on the agent acting as the
 * source of the transfer.
 */
public interface SourceTransferStartExit {

    /**
     * Invoked immediately prior to starting a transfer on the agent acting as
     * the source of the transfer.
     *
     * @param sourceAgentName
     *     the name of the agent acting as the source of the transfer.
     *     This is the name of the agent that the implementation of this
     *     method will be invoked from.
     *
     * @param destinationAgentName
     *     the name of the agent acting as the destination of the
     *     transfer.
     *
     * @param environmentMetaData
     *     meta data about the environment in which the implementation
     *     of this method is running. This information can only be read,
     *     it cannot be updated by the implementation. The constants
     *     defined in <code>EnvironmentMetaDataConstants</code> class can
     *     be used to access the data held by this map.
     *
     * @param transferMetaData
     *     meta data to associate with the transfer. The meta data passed
     *     to this method can be altered, and the changes to will be
     *     reflected in subsequent exit routine invocations. This map may
     *     also contain keys with IBM reserved names. These entries are
     *     defined in the <code>TransferMetaDataConstants</code> class and
     *     have special semantics.
     *
     * @param fileSpecs
     *     a list of file specifications that govern the file data to
     *     transfer. The implementation of this method can add entries,
     *     remove entries, or modify entries in this list and the changes
     *     will be reflected in the files transferred.
     *
     * @return
     *     a transfer exit result object which is used to determine if the
     *     transfer should proceed, or be cancelled.
     */
    TransferExitResult onSourceTransferStart(String sourceAgentName,
        String destinationAgentName,
        Map<String, String> environmentMetaData,
        Map<String, String> transferMetaData,
        List<SourceFileExitFileSpecification> fileSpecs);
}

```

Concepts associés

«Personnalisation de MFT avec des exits utilisateur», à la page 263

Vous pouvez personnaliser les fonctions d' Managed File Transfer à l'aide de vos propres programmes appelés routines d'exit utilisateur.

Référence associée

«SourceFileExitFileSpecification.java», à la page 936

«Interface SourceTransferEndExit.java», à la page 937

«Interface DestinationTransferStartExit.java», à la page 912

«Interface DestinationTransferEndExit.java», à la page 910

«Interface MonitorExit.java», à la page 931

«Interface ProtocolBridgeCredentialExit.java», à la page 932

Interface TransferExitResult.java

TransferExitResult.java

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2023. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */

package com.ibm.wmqfte.exitroutine.api;

/**
 * The result of invoking a transfer exit routine. It is composed of a result
 * code, which determines if the transfer should proceed, and an optional explanatory
 * message. The explanation, if present, is entered into the log message.
 */
public class TransferExitResult {

    private final TransferExitResultCode resultCode;
    private final String explanation;

    /**
     * For convenience, a static "proceed" result with no associated explanation
     * message.
     */
    public static final TransferExitResult PROCEED_RESULT =
        new TransferExitResult(TransferExitResultCode.PROCEED, null);

    /**
     * Constructor. Creates a transfer exit result object with a specified result
     * code and explanation.
     *
     * @param resultCode
     *     The result code to associate with the exit result being created.
     *
     * @param explanation
     *     The explanation to associate with the exit result being created.
     *     A value of <code>null</code> can be specified to indicate no
     *     explanation.
     */
    public TransferExitResult(TransferExitResultCode resultCode, String explanation) {
        this.resultCode = resultCode;
        this.explanation = explanation;
    }

    /**
     * Returns the explanation associated with this transfer exit result.
     *
     * @return
     *     the explanation associated with this exit result.
     */
    public String getExplanation() {
        return explanation;
    }

    /**
     * Returns the result code associated with this transfer exit result.
     *
     * @return
     *     the result code associated with this exit result.
     */
    public TransferExitResultCode getResultCode() {
        return resultCode;
    }
}
```

Référence associée

[«Interface SourceTransferStartExit.java»](#), à la page 938

[«Interface DestinationTransferStartExit.java»](#), à la page 912

[«Interface DestinationTransferEndExit.java»](#), à la page 910

[«Interface MonitorExit.java»](#), à la page 931

[«Interface ProtocolBridgeCredentialExit.java»](#), à la page 932

Information associée

[Personnalisation de MFT avec des exits utilisateur](#)

Formats de message pour les messages que vous pouvez placer dans la file d'attente de commandes de l'agent MFT

Les schémas XML suivants définissent les formats des messages qui peuvent être placés dans la file d'attente de commandes de l'agent pour demander à l'agent d'effectuer une action. Le message XML peut être placé dans la file d'attente de commandes de l'agent à l'aide des commandes de ligne de commande ou par une application.

Format de message de demande de transfert de fichier

Les transferts de fichiers sont initiés par les messages XML arrivant dans une file d'attente de commandes d'agent, généralement à la suite de l'émission d'une commande de transfert de fichiers par un utilisateur ou à l'aide de IBM MQ Explorer. Le XML de la demande de transfert doit être conforme au schéma `FileTransfer.xsd` et avoir l'élément `<request>` comme élément racine. Le document de schéma `FileTransfer.xsd` se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/schema`. Le `FileTransfer.xsd` schéma importe `fteutils.xsd`, qui se trouve dans le même répertoire.

Les messages de transfert de fichier peuvent comporter l'un des trois éléments racine suivants:

- `<request>`-pour les nouvelles demandes de transfert de fichiers, les demandes d'appels gérés ou la suppression de transferts planifiés en attente
- `<cancel>`-pour l'annulation des transferts de fichiers en cours
- `<transferSpecifications>`-pour spécifier plusieurs groupes de fichiers de transfert, utilisés par la commande **`fteCreateTransfer`**

Pour plus d'informations sur la spécification de plusieurs groupes de transfert à l'aide de l'élément `<transferSpecifications>`, voir [Utilisation de fichiers de définition de transfert](#).

Schéma

Le schéma suivant décrit les éléments valides dans un message XML de demande de transfert.

```
<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema">
<xsd:include schemaLocation="fteutils.xsd"/>

<!--
  Defines the request of a managed transfer and version number
  <request version="1.00" ...
    <managedTransfer>
      ...
    </managedTransfer>
  </request>
-->
<xsd:element name="request">
  <xsd:complexType>
    <xsd:choice>
      <xsd:element name="managedTransfer" type="managedTransferType"/>
      <xsd:element name="deleteScheduledTransfer" type="deleteScheduledTransferType"/>
      <xsd:element name="managedCall" type="managedCallType"/>
    </xsd:choice>
  </xsd:complexType>
</xsd:element>
```

```

        <xsd:attribute name="version" type="versionType" use="required"/>
    </xsd:complexType>
</xsd:element>

<!--
    Defines the cancel request of a managed transfer and version number
    <cancel version="1.00"
        xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
        xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
        <originator>
            <hostName>myMachine</hostName>
            <userID>myUserId</userID>
        </originator>      - Delete a scheduled transfer.

        <transfer>
            Transfer ID to Cancel
        </transfer>
    </cancel>
-->
<xsd:element name="cancel">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="originator" type="hostUserIDType" maxOccurs="1" minOccurs="1"/>
            <xsd:choice>
                <xsd:element name="transfer" type="IDType" maxOccurs="1" minOccurs="1"/>
                <xsd:element name="call" type="IDType" maxOccurs="1" minOccurs="1"/>
            </xsd:choice>
            <xsd:element name="reply" type="replyType" maxOccurs="1" minOccurs="0"/>
        </xsd:sequence>
        <xsd:attribute name="version" type="versionType" use="required"/>
    </xsd:complexType>
</xsd:element>

<!--
    Defines the transfer definition element structure.
    <transferSpecifications>
        <item ...
        <item ...
    </transferSpecifications>
-->
<xsd:element name="transferSpecifications">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="item" type="itemType" minOccurs="1" maxOccurs="unbounded"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

<!--
    Define a managed transfer of an instigator and request
    <managedTransfer>

        <originator>
            .
            .
            .
        </originator>

        <schedule>
            <submit timebase="source"|"UTC">2008-12-07T16:07</submit>
            <repeat>
                <frequency interval="hours">2</frequency>
                <expireTime>2008-12-0816:07</expireTime>
            </repeat>
        </schedule>

        <sourceAgent agent="here" QMgr="near"/>
        <destinationAgent agent="there" QMgr="far"/>

        <trigger>
            .
            .
            .
        </trigger>

        <transferSet>
            .
            .
            .
        </transferSet>
    </managedTransfer>
-->

    <xsd:complexType name="managedTransferType">
        <xsd:sequence>
            <xsd:element name="originator" type="origTransferRequestType" maxOccurs="1"
minOccurs="1"/>
            <xsd:element name="schedule" type="scheduleType" maxOccurs="1" minOccurs="0"/>

```

```

        <xsd:element name="sourceAgent" type="agentType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="destinationAgent" type="agentClientType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="trigger" type="triggerType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="reply" type="replyType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="transferSet" type="transferSetType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="job" type="jobType" maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>

<!--
    This is a modified form of origRequestType which is used on a managed transfer request.
    The hostName and userID are mandatory attributes in this case.
-->
<xsd:complexType name="origTransferRequestType">
    <xsd:sequence>
        <xsd:element name="hostName" type="xsd:string" minOccurs="1" maxOccurs="1"/>
        <xsd:element name="userID" type="xsd:string" minOccurs="1" maxOccurs="1"/>
        <xsd:element name="mqmdUserID" type="xsd:string" minOccurs="0" maxOccurs="1"/>
        <xsd:element name="webBrowser" type="xsd:string" minOccurs="0" maxOccurs="1"/>
        <xsd:element name="webUserID" type="xsd:string" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>

<!--
    Defines the transfer set of source and destination agent and one or more files
    <transferSet priority="1">
        <metaDataSet>
            <metaData key="keyname">keyvalue</metaData>
            <metaData key="keyname">keyvalue</metaData>
        </metaDataSet>

        <item>
            ...
        </item>
    </transferSet>
-->
<xsd:complexType name="transferSetType">
    <xsd:sequence>
        <xsd:element name="metaDataSet" type="metaDataSetType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="preSourceCall" type="commandActionType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="postSourceCall" type="commandActionType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="preDestinationCall" type="commandActionType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="postDestinationCall" type="commandActionType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="item" type="itemType" maxOccurs="unbounded" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="priority" type="priorityType" use="optional"/>
</xsd:complexType>

<!--
    Define a file pair with source and destination
    <item mode=[binary|text]>
        <source recursive="false" disposition="leave">
            <file>filename</file>
        </source>

        <destination type="file" exist="error">
            <file>filename</file>
        </destination>
    </item>
-->
<xsd:complexType name="itemType">
    <xsd:sequence>
        <xsd:element name="source" type="fileSourceType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="destination" type="fileDestinationType" maxOccurs="1" minOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute name="mode" type="modeType" use="required"/>
    <xsd:attribute name="checksumMethod" type="checkSumMethod" use="required"/>
</xsd:complexType>

<!--
    Defines the request to delete scheduled file transfer.
    <deleteScheduledTransfer>
        <originator>
            <delete>
                <hostName>myMachine</hostName>
                <userID>myUserId</userID>

```

```

        </delete>
    </originator>
    <ID>56</ID>
</deleteScheduledTransfer>
-->
<xsd:complexType name="deleteScheduledTransferType">
    <xsd:sequence>
        <xsd:element name="originator" type="origDeleteType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="ID" type="idType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="reply" type="replyType" maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="managedCallType">
    <xsd:sequence>
        <xsd:element name="originator" type="origRequestType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="agent" type="agentType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="reply" type="replyType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="transferSet" type="callTransferSetType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="job" type="jobType" maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="callTransferSetType">
    <xsd:sequence>
        <xsd:element name="metaDataSet" type="metaDataSetType" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="call" type="commandActionType" maxOccurs="1" minOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute name="priority" type="priorityType" use="optional"/>
</xsd:complexType>
</xsd:schema>

```

Description du message de demande de transfert

Les éléments et attributs utilisés dans les messages de demande de transfert sont décrits dans la liste suivante:

Descriptions des éléments

<request>

Élément de groupe contenant tous les éléments requis pour spécifier une demande de transfert de fichier.

Attribut	Description
version	Indique la version de cet élément fournie par Managed File Transfer.

<managedTransfer>

Élément de groupe contenant tous les éléments requis pour un transfert de fichier unique ou un groupe de transferts de fichier unique.

<TransfertdeleteScheduled>

Élément de groupe contenant les informations d'origine et d'ID permettant d'annuler un transfert de planification.

<managedCall>

Élément de groupe qui contient tous les éléments requis pour un seul appel géré d'un programme ou d'un exécutable.

< ID >

Identificateur unique qui spécifie la demande de transfert à supprimer de la liste des transferts planifiés en attente.

< originaire >

Élément de groupe qui contient les éléments spécifiant l'émetteur de la demande.

<hostName>

Nom d'hôte du système sur lequel se trouve le fichier source.

<userID>

ID utilisateur à l'origine du transfert de fichiers.

< IDmqmdUser>

Facultatif. ID utilisateur IBM MQ fourni dans le descripteur de message (MQMD).

< planification>

Élément de groupe décrivant l'heure planifiée pour le transfert de fichier, le comportement de répétition et la date d'échéance de la prochaine occurrence.

< soumettre>

Indique la date et l'heure auxquelles le transfert planifié doit commencer.

Attribut	Description
base de temps	Indique le fuseau horaire à utiliser. Cet attribut peut adopter l'une des valeurs suivantes : <ul style="list-style-type: none"> • source-utiliser le fuseau horaire de l'agent source • admin-utilisez le fuseau horaire de l'administrateur qui a émis la commande • UTC-utiliser le temps universel coordonné
fuseau horaire	Description du fuseau horaire en fonction de la valeur de base de temps

< répétition>

Élément de groupe qui contient des détails sur la fréquence de répétition d'un transfert planifié, le nombre de répétitions d'un transfert planifié et le moment où un transfert planifié cesse de se répéter.

< fréquence>

Période qui doit s'écouler avant que le transfert ne se répète.

Attribut	Description
intervalle	Les unités d'intervalle, qui doivent être l'une des valeurs suivantes: <ul style="list-style-type: none"> • minutes • heures • jours • semaines • mois • années

<expireTime>

Élément facultatif indiquant la date et l'heure d'arrêt d'un transfert planifié répétitif. Cet élément et l'élément <expireCount> s'excluent mutuellement.

<expireCount>

Élément facultatif qui indique le nombre de fois où le transfert de fichiers planifié se produit avant l'arrêt. Cet élément et l'élément <expireTime> s'excluent mutuellement.

<sourceAgent>

Indique le nom de l'agent sur le système où se trouve le fichier source.

Attribut	Description
Agent	Indique le nom de l'agent.
QMgr	Nom du gestionnaire de files d'attente d'agent.

<destinationAgent>

Indique le nom de l'agent sur le système vers lequel vous souhaitez transférer le fichier.

Attribut	Description
Agent	Indique le nom de l'agent.
QMgr	Nom du gestionnaire de files d'attente d'agent.
hostName	Nom d'hôte ou adresse IP du gestionnaire de files d'attente de l'agent.
portNumber	Numéro de port utilisé pour les connexions client au gestionnaire de files d'attente de l'agent de destination.
canal	Nom de canal utilisé pour la connexion au gestionnaire de files d'attente de l'agent de destination.

< déclencheur>

Élément facultatif qui spécifie une condition qui doit être vraie pour que le transfert de fichier ait lieu.

Attribut	Description
journal	Indicateur signalant si les échecs de déclenchement sont consignés. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> • yes-des entrées de journal sont créées pour les transferts déclenchés ayant échoué • no-les entrées de journal ne sont pas créées pour les transferts déclenchés ayant échoué

<fileExist>

Indique une liste séparée par des virgules de noms de fichiers situés sur le même système que l'agent source. Si un fichier de cette liste de noms remplit la condition du déclencheur, le transfert est effectué. Cet élément et l'élément <fileSize> s'excluent mutuellement.

Attribut	Description
comparaison	Indique comment évaluer les noms de fichier source par rapport à la liste de noms. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> • = au moins un nom de fichier dans la liste de noms doit correspondre • != un minimum de l'un des fichiers de la liste de noms n'existe pas
valeur	Indique le type de comparaison: <ul style="list-style-type: none"> • exist: le fichier doit exister

<fileSize>

Indique une liste séparée par des virgules de noms de fichiers situés sur le même système que l'agent source. Si un fichier de cette liste de noms remplit la condition du déclencheur, le transfert est effectué. Cet élément et l'élément <fileExist> s'excluent mutuellement.

Attribut	Description
comparaison	Indique comment évaluer les noms de fichier source par rapport à la liste de noms. La valeur valide est la suivante:

Attribut	Description
	<ul style="list-style-type: none"> • > = l'un des noms de fichier de la liste de noms existe et a une taille minimale spécifiée dans l'attribut de valeur
valeur	Taille de fichier indiquée sous la forme d'un entier avec les unités suivantes: <ul style="list-style-type: none"> • B-octets • Ko-kilooctets • Mo-mégaoctets • Giga-Go (la valeur des unités n'est pas sensible à la casse)

< réponse >

Indique le nom de la file d'attente de réponses temporaire générée pour les transferts de fichiers synchrones (indiquée avec le paramètre **-w** sur la ligne de commande). Le nom de la file d'attente est défini par la clé **dynamicQueuePrefix** dans le fichier de configuration `command.properties` ou par la valeur par défaut `WMQFTE.*` Si aucune valeur n'est spécifiée,

Attribut	Description
détaillé	Indique si des informations détaillées sur les résultats de transfert sont requises dans le message de réponse. Plusieurs messages de réponse peuvent être générés pour chaque transfert. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> • true-des informations de réponse détaillées sont requises. Le format des informations est le même que celui publié dans le journal de transfert dans les messages de progression, c'est-à-dire l'élément <transferSet>. Pour plus d'informations, voir la section «Formats des messages du journal de transfert de fichiers», à la page 771. Des informations de réponse détaillées sont disponibles uniquement lorsque la propriété <code>enableDetailedReplyMessages</code> est définie sur true pour l'agent source de transfert. • false-les informations de réponse détaillées ne sont pas requises. La valeur par défaut est false.
QMGR	Nom du gestionnaire de files d'attente de commandes sur lequel la file d'attente dynamique temporaire est générée pour recevoir les réponses.
persistant	Indique si le message écrit dans la file d'attente de réponses est persistant. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> • true-le message est persistant • false-le message n'est pas persistant • qdef-la persistance du message est définie par les propriétés de la file d'attente de réponses La valeur par défaut est false.

<transferSet>

Indique un groupe de transferts de fichiers que vous souhaitez effectuer ensemble ou un groupe d'appels gérés que vous souhaitez effectuer ensemble. Lors de la transmission, <transferSet> est un élément de groupe contenant des éléments <item>.

Attribut	Description
priorité	Niveau de priorité du transfert. La priorité est une valeur comprise entre 0 et 9, où 0 est la priorité la plus basse. Le niveau de priorité par défaut est 0 et, par défaut, le transfert utilise le niveau de priorité de l'agent source.

< Ensemble de métadonnéesmetaData>

Élément de groupe facultatif contenant un ou plusieurs éléments de métadonnées.

<metaData>

Indique les métadonnées définies par l'utilisateur qui sont transmises aux points d'exit appelés par l'agent. L'élément contient la valeur de métadonnées sous forme de chaîne.

Attribut	Description
clé	Nom de métadonnées sous forme de chaîne

< appel>

Élément de groupe qui contient des éléments < command> spécifiant le programme ou l'exécutable à appeler.

< commande>

Indique le programme ou l'exécutable à appeler. La commande doit se trouver dans le chemin de la commande de l'agent. Pour plus d'informations, voir [Tableau 45, à la page 566](#). Cet élément peut contenir des éléments facultatifs < argument>.

Attribut	Description
nom	Nom de la commande.
successRC	Code retour de réussite renvoyé par cette commande. La valeur par défaut est 0.
retryCount	Nombre de tentatives de relance de la commande en cas d'échec.
retryWait	Délai d'attente, en secondes, entre les nouvelles tentatives de la commande.
type	Type de programme à appeler. Les valeurs valides sont antscript, jcl ou exécutable.

<argument>

Indique un argument à transmettre à la commande.

<item>

Élément de groupe qui contient des éléments spécifiant les noms et les emplacements des fichiers source et de destination.

Attribut	Description
mode	Indique que le mode de transfert est binaire ou texte.
checksumMethod	Indique le type d'algorithme de hachage qui génère le résumé du message pour créer la signature numérique. Les valeurs valides sont MD5 ou aucune.

< source>

Élément de groupe qui spécifie les fichiers sur le système source et s'ils sont supprimés à la fin du transfert

Attribut	Description
recursive	Indique que les fichiers sont transférés de manière récursive dans des sous-répertoires lorsque l'élément < source> est un répertoire ou contient des caractères génériques.
disposition	Indique l'action effectuée sur l'élément < source> lorsque < source> a été transféré vers sa destination. Les valeurs admises sont les suivantes : <ul style="list-style-type: none">• leave-les fichiers source restent inchangés.• delete-les fichiers source sont supprimés du système source après le transfert du fichier source.

<fichier>

Indique la source du transfert. **Multi** Pour Multiplatforms, la source de transfert peut être un fichier ou un nom de répertoire. Pour la plateforme z/OS, la source de transfert peut être un nom de fichier, de répertoire, de fichier ou de fichier partitionné. Utilisez le chemin qualifié complet dans le format correspondant à votre système d'exploitation, par exemple C:/from/here.txt. N'utilisez pas d'URI de fichier.

Attribut	Description
alias	Indique un alias pour le fichier source. Cet alias est le nom du fichier source, à l'exclusion de tout chemin de répertoire spécifié pour le transfert.
EOL	Indique le marqueur de fin de ligne pour les transferts de texte. Les valeurs admises sont : <ul style="list-style-type: none">• Saut de ligne-caractère de saut de ligne uniquement• CRLF-retour chariot et séquence de caractères de saut de ligne
codage	Codage du fichier source pour un transfert de fichier texte.
z/OS délimiteur	Indique le délimiteur qui est inclus entre les enregistrements dans les fichiers source orientés enregistrement, par exemple, les fichiers z/OS. Indiquez la valeur de délimiteur sous la forme de deux chiffres hexadécimaux dans la plage 00-FF, avec le préfixe x. Par exemple, x12 ou x03,x7F.
delimiterType	Indique le type de délimiteur inclus dans le fichier de destination après les données de message individuelles. Les valeurs valides sont les suivantes: <ul style="list-style-type: none">• binary-délimiteur hexadécimal Cet attribut est disponible uniquement si vous avez activé la fonction 7.0.4.1.
delimiterPosition	Indique la position à laquelle insérer des délimiteurs lors de l'écriture d'enregistrements de fichier source orientés enregistrement dans un fichier normal. Les valeurs admises sont les suivantes : <ul style="list-style-type: none">• prefix-le délimiteur est inséré dans le fichier de destination avant les données de chaque enregistrement de fichier orienté enregistrement source.• postfix-le délimiteur est inséré dans le fichier de destination après les données de chaque enregistrement de fichier orienté enregistrement source.
includeDelimiterInFile	Indique s'il faut inclure un délimiteur entre les enregistrements dans les fichiers source orientés enregistrement.
z/OS Espaces keepTrailing	Indique si les espaces de fin doivent être conservés dans les enregistrements source lus à partir d'un fichier de longueur fixe dans le cadre d'un transfert en mode texte. Par défaut, les espaces de fin sont supprimés. Les valeurs admises sont les suivantes : <ul style="list-style-type: none">• true-les espaces de fin sont conservés dans les enregistrements source lus à partir d'un ensemble de données de format de longueur fixe• false-les espaces de fin sont supprimés des enregistrements source lus à partir d'un ensemble de données de format de longueur fixe

< file_attente >

Lorsqu'il est utilisé avec l'élément < source >, indique le nom de la file d'attente à partir de laquelle effectuer le transfert, qui doit se trouver sur le gestionnaire de files d'attente de l'agent source. Utilisez le format *QUEUE*. N'incluez pas le nom du gestionnaire de files d'attente, la file d'attente doit être présente sur le gestionnaire de files d'attente de l'agent source. Vous ne pouvez pas utiliser l'élément < queue > dans l'élément < source >, si vous l'avez utilisé dans l'élément < destination >.

Attribut	Description
useGroups	Indique s'il faut transférer uniquement le premier groupe complet de messages de la file d'attente source. Les valeurs admises sont les suivantes : <ul style="list-style-type: none">• true-transfère uniquement le premier groupe complet de messages• false-transfère tous les messages dans la file d'attente source
groupId	Indique le groupe de messages à lire dans la file d'attente source. Cet attribut est valide uniquement lorsque la valeur de l'attribut useGroups est true.
delimiterType	Indique le type de délimiteur inclus dans le fichier de destination après les données de message individuelles. Les valeurs admises sont les suivantes : <ul style="list-style-type: none">• text-délimiteur de texte ou de littéral Java• binary-délimiteur hexadécimal
délimiteur	Indique le délimiteur inclus dans le fichier de destination entre les données de message individuelles.
delimiterPosition	Indique si le délimiteur est inclus dans le fichier de destination avant ou après les données de message individuelles. Les valeurs admises sont les suivantes : <ul style="list-style-type: none">• prefix-le délimiteur est inclus avant les données• postfix-le délimiteur est inclus après les données
codage	Indique le codage de la file d'attente source.
waitTime	Indique le délai d'attente, en secondes, de l'agent source pour l'un des éléments suivants: <ul style="list-style-type: none">• un message à afficher dans la file d'attente source, si la file d'attente est vide ou est devenue vide• un groupe complet à afficher dans la file d'attente source, si l'attribut useGroups a été défini sur true <p>Pour plus d'informations sur la définition de la valeur waitTime , voir «Conseils pour la spécification d'un temps d'attente sur un transfert de message vers fichier», à la page 704.</p>

< destination >

Élément de groupe qui spécifie la destination et le comportement si des fichiers existent au niveau de l'agent de destination.

Vous ne pouvez spécifier qu'un seul élément < file > et < queue > comme élément enfant de la destination.

Attribut	Description
type	Type de destination. Les valeurs admises sont les suivantes :

Attribut	Description
	<ul style="list-style-type: none"> • <code>file</code> -indique un fichier comme destination • <code>directory</code> -indique un répertoire comme destination • <code>> z/OS dataset</code> -spécifie un fichier z/OS comme destination • <code>> z/OS pds</code> -indique un fichier partitionné z/OS comme destination • <code>queue</code> -indique une file d'attente IBM MQ comme destination • <code>filespace</code> -indique un espace fichier comme destination <p>La valeur <code>queue</code> est valide uniquement lorsque l'élément < destination > a un élément enfant < queue >.</p> <p>La valeur <code>filespace</code> est valide uniquement lorsque l'élément < destination > a un élément enfant de < filespace >.</p> <p>Les autres valeurs sont valides uniquement lorsque l'élément < destination > a un élément enfant < fichier >.</p>
existe	<p>Indique l'action effectuée si un fichier de destination existe sur le système de destination. Les valeurs admises sont les suivantes :</p> <ul style="list-style-type: none"> • <code>error</code>-signale une erreur et le fichier n'est pas transféré. • <code>overwrite</code>-écrase le fichier de destination existant. <p>Cet attribut n'est pas valide si l'élément < destination > a un élément enfant < queue > ou < filespace >.</p>

<fichier>

Spécifie des paramètres supplémentaires pour l'élément **< destination >** décrit précédemment. Utilisez le chemin qualifié complet dans le format correspondant à votre système d'exploitation, par exemple `C:/from/here.txt`. N'utilisez pas d'URI de fichier.

Attribut	Description
alias	Indique un alias pour le fichier < destination > . Cet alias est le nom du fichier source, à l'exclusion de tout chemin de répertoire spécifié pour le transfert.
codage	Codage du fichier < destination > pour un transfert de fichier texte.
EOL	Indique le marqueur de fin de ligne pour les transferts de texte. Les valeurs admises sont : <ul style="list-style-type: none"> • Saut de ligne-caractère de saut de ligne uniquement • CRLF-retour chariot et séquence de caractères de saut de ligne
truncateRecords	Facultatif. Indique que les enregistrements < destination > plus longs que l'attribut de fichier LRECL sont tronqués. <ul style="list-style-type: none"> • <code>True</code>-les enregistrements sont tronqués • <code>False</code>-les enregistrements sont encapsulés <p>Par défaut, le paramètre a la valeur <code>false</code>.</p>

< file_attente >

Lorsqu'il est utilisé avec l'élément **< destination >** , indique le nom de la file d'attente vers laquelle effectuer le transfert, qui peut se trouver sur tout gestionnaire de files d'attente connecté au gestionnaire de files d'attente de l'agent de destination. Utilisez le format `QUEUE@QM` où `QUEUE` est le nom de la file d'attente dans laquelle placer les messages et `QM` est le gestionnaire de files

d'attente dans lequel se trouve la file d'attente. Vous ne pouvez pas utiliser l'élément < queue> dans l'élément < destination>, si vous l'avez utilisé dans l'élément < source>.

Attribut	Description
délimiteur	Délimiteur permettant de fractionner le fichier en plusieurs messages.
delimiterType	Indique le type de délimiteur. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> • text-expression régulière Java • binary-une séquence d'octets hexadécimaux • size-nombre d'octets, de kibioctets ou de mébioctets. Par exemple, 1 B, 1 K ou 1 M.
delimiterPosition	Indique si le délimiteur est attendu avant ou après les données à inclure dans les messages individuels. Les options valides sont les suivantes: <ul style="list-style-type: none"> • prefix-le délimiteur est attendu avant les données • postfix-le délimiteur est attendu après les données
includeDelimiterInMessage	Valeur booléenne indiquant s'il faut inclure les délimiteurs utilisés pour fractionner le fichier en plusieurs messages à la fin des messages.
codage	Indique le codage de la file d'attente de destination.
persistant	Indique si les messages sont persistants. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> • true-les messages sont persistants • false-les messages ne sont pas persistants • qdef-la valeur de persistance des messages est définie par les paramètres de la file d'attente de destination
setMqProps	Valeur booléenne indiquant si les propriétés de message IBM MQ sont définies sur le premier message d'un fichier et sur tous les messages écrits dans la file d'attente lorsqu'une erreur se produit.
Page unrecognizedCode	Indique si un transfert en mode texte échoue ou si la conversion est effectuée si la page de codes des données n'est pas reconnue par le gestionnaire de files d'attente de destination. Les valeurs admises sont les suivantes : <ul style="list-style-type: none"> • fail-le transfert signale un échec • binary-les données sont converties dans la page de codes de destination et l'en-tête de message IBM MQ décrivant le format des données est défini sur MQFMT_NONE. <p>Le comportement par défaut est fail.</p>

< espace_fichier>

Élément de groupe spécifiant le nom de l'espace fichier vers lequel effectuer le transfert.

< nom>

Lorsqu'il est utilisé avec l'élément < filespace>, la valeur de cet élément indique le nom de l'espace fichier.

< appelpreSource>

Élément de groupe spécifiant une commande à appeler à la source du transfert, avant le début du transfert.

< appelpostSource >

Élément de groupe spécifiant une commande à appeler à la source du transfert, une fois le transfert terminé.

< appelpreDestination >

Élément de groupe spécifiant une commande à appeler à la destination du transfert, avant le début du transfert.

< appelpostDestination >

Élément de groupe spécifiant une commande à appeler à la destination du transfert, une fois le transfert terminé.

< commande >

Lorsqu'il est utilisé avec l'élément <preSourceCall>, <postSourceCall>, <preDestinationCall> ou <postDestinationCall >, cet élément spécifie la commande à appeler. La commande doit se trouver dans le chemin de la commande de l'agent. Pour plus d'informations, voir [Tableau 45](#), à la page 566.

Attribut	Description
nom	Nom de la commande à exécuter.
successRC	Code retour attendu si la commande s'exécute correctement.

< argument >

Lorsqu'il est utilisé avec l'élément < command >, cet élément spécifie un argument à transmettre à la commande. Vous pouvez avoir n'importe quel nombre d'éléments < argument > dans un élément < command >.

< travail >

Élément de groupe facultatif contenant des informations de travail pour l'ensemble de la spécification de transfert. < job > est un identificateur de nom de travail défini par l'utilisateur qui est ajouté au message du journal lorsque le transfert a commencé. Cet élément < job > est identique à l'élément < job > qui apparaît dans le message du journal de transfert, décrit dans la rubrique suivante: [«Formats des messages du journal de transfert de fichiers»](#), à la page 771.

< nom >

Lorsqu'il est utilisé avec l'élément < job >, la valeur de cet élément indique le nom du travail.

< transferSpecifications >

Élément de groupe qui contient des éléments < item > pour plusieurs groupes de transfert. Pour plus d'informations sur l'utilisation de cet élément, voir [Utilisation des fichiers de définition de transfert](#) .

< annuler >

Élément de groupe contenant tous les éléments requis pour annuler un transfert de fichier en cours.

Attribut	Description
version	Indique la version de cet élément fournie par Managed File Transfer.

< transfer >

Lorsqu'il est utilisé avec l'élément < cancel >, la valeur de cet élément indique l'ID de demande de transfert à annuler.

< travail >

Élément de groupe contenant des informations sur le travail.

< jobName >

Indique l'identificateur de travail logique.

Format du message d'annulation du transfert de fichiers

Une demande de transfert de fichier renvoie un ID de 48 caractères qui identifie le transfert pour un agent spécifique. Cet ID est utilisé pour annuler les transferts.

Description du message d'annulation de transfert

Les éléments et attributs utilisés dans les messages d'annulation de transfert sont décrits ci-dessous:

< annuler >

Élément de groupe contenant tous les éléments requis pour annuler un transfert de fichier en cours.

Attribut	Description
version	Indique la version de cet élément fournie par Managed File Transfer.

< originaire >

Élément de groupe qui contient les éléments spécifiant l'émetteur de la demande.

< hostName >

Nom d'hôte du système sur lequel se trouve le fichier source.

< userID >

ID utilisateur à l'origine du transfert de fichiers.

< IDmqmdUser >

Facultatif. ID utilisateur IBM MQ fourni dans le descripteur de message (MQMD).

< transfer >

Lorsqu'il est utilisé avec l'élément < cancel >, la valeur de cet élément indique l'ID de demande de transfert à annuler.

< travail >

Facultatif. Élément de groupe contenant des informations sur le travail.

< jobName >

Indique l'identificateur de travail logique.

Exemples

Des exemples de messages XML conformes à ce schéma sont fournis pour chacune des demandes suivantes:

- [Créer un transfert de fichier](#)
- [Création d'une demande de transfert de fichier asynchrone](#)
- [Annuler un transfert de fichier](#)
- [Création d'un transfert planifié](#)
- [Suppression d'un transfert planifié](#)
- [Création d'un appel géré](#)
- [Création d'un transfert de fichiers incluant des appels gérés](#)

Exemples de message de demande de transfert de fichier

Exemples de messages que vous pouvez placer dans la file d'attente de commandes de l'agent pour demander à l'agent de créer ou d'annuler un transfert.

Créer une demande de transfert

```
<?xml version="1.0" encoding="UTF-8"?>
<request xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
version="4.00"
xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>example.com.</hostName>
      <userID>mqm</userID>
    </originator>
    <sourceAgent QMgr="QM_JUPITER" agent="AGENT_JUPITER"/>
    <destinationAgent QMgr="QM_JUPITER" agent="AGENT_JUPITER"/>
    <transferSet>
      <item checksumMethod="MD5" mode="binary">
        <source disposition="leave" recursive="false">
          <file>/etc/passwd</file>
        </source>
        <destination exist="overwrite" type="directory">
          <file>/tmp</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Créer une demande de transfert-synchrone

Lorsqu'un utilisateur demande une demande synchrone de blocage, c'est-à-dire qu'il attend la fin du transfert et qu'il reçoit des messages d'état, le message placé dans la file d'attente de commandes contient un élément de réponse qui spécifie la file d'attente à laquelle un message de réponse est envoyé. L'exemple suivant illustre le message placé dans la file d'attente de commandes utilisée par FTEAGENT:

```
<?xml version="1.0" encoding="UTF-8"?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="FTEAGENT"
      QMgr="QM1"/>
    <destinationAgent agent="AGENT2"
      QMgr="QM2"/>
    <reply QMGR="QM1">WMQFTE.492D0D5502770020</reply>
    <transferSet>
      <item mode="binary" checksumMethod="MD5">
        <source recursive="false" disposition="leave">
          <file>c:\sourcefiles\source1.doc</file>
        </source>
        <destination type="file" exist="overwrite">
          <file>c:\destinationfiles\dest1.doc</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

L'élément < reply> est rempli avec le nom du gestionnaire de files d'attente de commandes dans lequel une file d'attente dynamique temporaire a été créée pour recevoir une réponse indiquant que le transfert a abouti (ou non). Le nom de la file d'attente dynamique temporaire se compose de deux parties:

- Préfixe tel que défini par la clé **dynamicQueuePrefix** dans le fichier de configuration `command.properties` (il s'agit de WMQFTE). par défaut)
- ID de la file d'attente généré par IBM MQ

Demande d'annulation de transfert

```
<?xml version="1.0" encoding="UTF-8"?>
<cancel xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
  </originator>
  <transfer>414D51205553322E42494E444494E47538B0F404D032C0020</transfer>
  <reply QMGR="QM_JUPITER">WMQFTE.4D400F8B20002007</reply>
</cancel>
```

Référence associée

«Format de message de demande de transfert de fichier», à la page 804

Les transferts de fichiers sont initiés par les messages XML arrivant dans une file d'attente de commandes d'agent, généralement à la suite de l'émission d'une commande de transfert de fichiers par un utilisateur ou à l'aide de IBM MQ Explorer. Le XML de la demande de transfert doit être conforme au schéma FileTransfer.xsd et avoir l'élément <request> comme élément racine. Le document de schéma FileTransfer.xsd se trouve dans le répertoire MQ_INSTALLATION_PATH/mqft/samples/schema. Le FileTransfer.xsd schéma importe fteutils.xsd, qui se trouve dans le même répertoire.

Exemples de message de transfert de fichier planifié

Exemples de messages que vous pouvez placer dans la file d'attente de commandes de l'agent pour demander à l'agent de créer ou de supprimer une planification.

Créer un transfert planifié

```
<?xml version="1.0" encoding="UTF-8"?>
<request xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>example.com.</hostName>
      <userID>mqm</userID>
    </originator>
    <schedule>
      <submit timebase="admin" timezone="Europe/London">2010-01-01T21:00</submit>
    </schedule>
    <sourceAgent QMgr="US2.BINDINGS" agent="US2.BINDINGS.FILE"/>
    <destinationAgent QMgr="US2.BINDINGS" agent="US2.BINDINGS.FILE"/>
    <transferSet>
      <item checksumMethod="MD5" mode="binary">
        <source disposition="leave" recursive="false">
          <file>/etc/passwd</file>
        </source>
        <destination exist="overwrite" type="directory">
          <file>/tmp</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Supprimer le transfert planifié

```
<?xml version="1.0" encoding="UTF-8"?>
<request xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="4.00"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <deleteScheduledTransfer>
    <originator>
      <delete>
        <hostName>example.com.</hostName>
        <userID>mqm</userID>
      </delete>
    </originator>
```

```

    <ID>1</ID>
    <reply QMGR="US2.BINDINGS">WMQFTE.4D400F8B20003902</reply>
  </deleteScheduledTransfer>
</request>

```

Référence associée

«Format de message de demande de transfert de fichier», à la page 804

Les transferts de fichiers sont initiés par les messages XML arrivant dans une file d'attente de commandes d'agent, généralement à la suite de l'émission d'une commande de transfert de fichiers par un utilisateur ou à l'aide de IBM MQ Explorer. Le XML de la demande de transfert doit être conforme au schéma `FileTransfer.xsd` et avoir l'élément `<request>` comme élément racine. Le document de schéma `FileTransfer.xsd` se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/schema`. Le `FileTransfer.xsd` schéma importe `fteutils.xsd`, qui se trouve dans le même répertoire.

Exemples de message de demande d'appel d'agent MFT

Exemples de messages que vous pouvez placer dans la file d'attente de commandes de l'agent pour demander à l'agent de créer un appel géré ou un transfert qui appelle des programmes.

Exemple de demande d'appel géré

```

<?xml version="1.0" encoding="UTF-8"?>
<request xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="1.00"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedCall>
    <originator>
      <hostName>example.com.</hostName>
      <userID>mqm</userID>
    </originator>
    <agent agent="DNWE" QMgr="QM1"/>
    <transferSet>
      <call>
        <command name="echo" successRC="0">
          <argument>call</argument>
          <argument>test</argument>
        </command>
      </call>
    </transferSet>
    <job>
      <name>managedCallCalls.xml</name>
    </job>
  </managedCall>
</request>

```

Exemple de demande de transfert géré avec des appels

```

<?xml version="1.0" encoding="UTF-8"?>
<request xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="1.00"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>example.com.</hostName>
      <userID>mqm</userID>
    </originator>
    <sourceAgent agent="DNWE" QMgr="QM1"/>
    <destinationAgent agent="DNWE" QMgr="QM1"/>
    <transferSet>
      <preSourceCall>
        <command name="echo" successRC="0">
          <argument>preSourceCall</argument>
          <argument>test</argument>
        </command>
      </preSourceCall>
      <postSourceCall>
        <command name="echo" successRC="0">
          <argument>postSourceCall</argument>
          <argument>test</argument>
        </command>
      </postSourceCall>
    </transferSet>
  </managedTransfer>

```

```

    <preDestinationCall>
      <command name="echo" successRC="0">
        <argument>preDestinationCall</argument>
        <argument>test</argument>
      </command>
    </preDestinationCall>
    <postDestinationCall>
      <command name="echo" successRC="0">
        <argument>postDestinationCall</argument>
        <argument>test</argument>
      </command>
    </postDestinationCall>
  </transferSet>
</job>
  <name>managedTransferCalls.xml</name>
</job>
</managedTransfer>
</request>

```

Concepts associés

«[Spécification des programmes à exécuter avec MFT](#)», à la page 256

Vous pouvez exécuter des programmes sur un système sur lequel un Managed File Transfer Agent est en cours d'exécution. Dans le cadre d'une demande de transfert de fichiers, vous pouvez spécifier un programme à exécuter avant ou après le début d'un transfert. En outre, vous pouvez démarrer un programme qui ne fait pas partie d'une demande de transfert de fichier en soumettant une demande d'appel géré.

Référence associée

«[Format de message de demande de transfert de fichier](#)», à la page 804

Les transferts de fichiers sont initiés par les messages XML arrivant dans une file d'attente de commandes d'agent, généralement à la suite de l'émission d'une commande de transfert de fichiers par un utilisateur ou à l'aide de IBM MQ Explorer. Le XML de la demande de transfert doit être conforme au schéma `FileTransfer.xsd` et avoir l'élément `<request>` comme élément racine. Le document de schéma `FileTransfer.xsd` se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/schema`. Le `FileTransfer.xsd` schéma importe `fteutils.xsd`, qui se trouve dans le même répertoire.

Formats des messages de demande du moniteur MFT

Les moniteurs de ressources sont créés lorsqu'un message XML approprié arrive dans la file d'attente de commandes d'un agent, généralement suite à l'émission par un utilisateur de la commande de surveillance `fteCreateou` à l'utilisation de l'interface IBM MQ Explorer.

Le code XML du moniteur doit être conforme au schéma `Monitor.xsd` en utilisant l'élément `<monitor>` comme élément racine.

Les messages du moniteur peuvent comporter l'un des éléments racine suivants:

- `<monitor>` -pour la création et le démarrage d'un nouveau moniteur de ressources
- `<deleteMonitor>` -pour l'arrêt et la suppression d'un moniteur existant

Il n'y a pas de message de commande pour la commande `fteListMonitors` car la commande extrait directement les définitions de moniteur correspondantes de `SYSTEM.FTE`.

Schéma

Le schéma suivant décrit les éléments valides dans un message XML de demande de moniteur.

```

<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema"
  targetNamespace="https://www.ibm.com/xmlns/wmqfte/7.0.1/
  MonitorDefinition"
  xmlns="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition">
  <xsd:include schemaLocation="FileTransfer.xsd"/>
  <xsd:element name="monitor">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="name" type="monitorNameType"
          minOccurs="1" maxOccurs="1"/>

```

```

        <xsd:element name="description" type="xsd:string"
        minOccurs="0" maxOccurs="1"/>
        <xsd:element name="pollInterval" type="pollIntervalType"
        minOccurs="1" maxOccurs="1"
default="10"/>
        <xsd:element name="batch" type="batchType"
        minOccurs="0" maxOccurs="1"/>
        <xsd:element name="agent" type="agentNameType"
        minOccurs="1" maxOccurs="1"/>
        <xsd:element name="resources" type="monitorResourcesType"
        minOccurs="0"
maxOccurs="1"/>
        <xsd:element name="triggerMatch" type="triggerMatchType"
        maxOccurs="1" minOccurs="1"/>
        <xsd:element name="reply" type="replyType"
        maxOccurs="1" minOccurs="0"/>
        <xsd:element name="tasks" type="monitorTasksType"
        maxOccurs="1" minOccurs="1"/>
        <xsd:element name="originator" type="origRequestType"
        maxOccurs="1" minOccurs="1"/>
        <xsd:element name="job" type="jobType"
        maxOccurs="1" minOccurs="0"/>
        <xsd:element name="defaultVariables" type="defaultVariablesType"
        maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="version" type="versionType" use="required"/>
</xsd:complexType>
</xsd:element>

<xsd:element name="deleteMonitor">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="name" type="monitorNameType"
            minOccurs="1" maxOccurs="1"/>
            <xsd:element name="originator" type="origRequestType"
            maxOccurs="1" minOccurs="1"/>
            <xsd:element name="reply" type="replyType"
            maxOccurs="1" minOccurs="0"/>
        </xsd:sequence>
        <xsd:attribute name="version" type="versionType" use="required"/>
    </xsd:complexType>
</xsd:element>

<xsd:complexType name="transferRequestType">
    <xsd:choice>
        <xsd:element name="managedTransfer" type="managedTransferType"/>
        <xsd:element name="managedCall" type="managedCallType"/>
    </xsd:choice>
    <xsd:attribute name="version" type="versionType"/>
</xsd:complexType>

<xsd:complexType name="monitorResourcesType">
    <xsd:choice>
        <xsd:sequence>
            <xsd:element name="directory" type="monitoredDirectoryType"
            minOccurs="1" maxOccurs="unbounded"/>
        </xsd:sequence>
        <xsd:element name="queue" type="monitoredQueueType"/>
    </xsd:choice>
</xsd:complexType>

<xsd:complexType name="monitoredDirectoryType">
    <xsd:simpleContent>
        <xsd:extension base="xsd:string">
            <xsd:attribute name="recursionLevel"
type="xsd:nonNegativeInteger"/>
            <xsd:attribute name="id" type="resourceIdAttrType"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="monitoredQueueType">
    <xsd:simpleContent>
        <xsd:extension base="xsd:string">
            <xsd:attribute name="id" type="resourceIdAttrType"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="triggerMatchType">
    <xsd:sequence>
        <xsd:element name="conditions" type="conditionsType"

```

```

        minOccurs="1"      maxOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="conditionsType">
    <xsd:choice minOccurs="1">
        <xsd:element name="allOf"      type="listPredicateType"
            minOccurs="1"      maxOccurs="1"/>
        <xsd:element name="anyOf"      type="listPredicateType"
            minOccurs="1"      maxOccurs="1"/>
        <xsd:element name="condition" type="conditionType"
            minOccurs="1"      maxOccurs="1"/>
    </xsd:choice>
</xsd:complexType>

<xsd:complexType name="listPredicateType">
    <xsd:choice>
        <xsd:element name="condition" type="conditionType"
            minOccurs="1"      maxOccurs="unbounded"/>
    </xsd:choice>
</xsd:complexType>

<xsd:complexType name="conditionType">
    <xsd:sequence>
        <xsd:element name="name"      type="conditionNameType"
            minOccurs="0"      maxOccurs="1"/>
        <xsd:element name="resource" type="resourceIdType"
            minOccurs="0"      maxOccurs="1"/>
        <xsd:choice minOccurs="1">
            <xsd:element name="fileMatch"
type="fileMatchConditionType"
                minOccurs="1"      maxOccurs="1"/>
            <xsd:element name="fileNoMatch"
type="fileNoMatchConditionType"
                minOccurs="1"
maxOccurs="1"/>
            <xsd:element name="fileSize"
type="fileSizeConditionType"
                minOccurs="1"      maxOccurs="1"/>
            <xsd:element name="queueNotEmpty"
type="queueNotEmptyConditionType"
                minOccurs="1"      maxOccurs="1"/>
            <xsd:element name="completeGroups"
type="completeGroupsConditionType"
                minOccurs="1"      maxOccurs="1"/>
            <xsd:element name="fileSizeSame"
type="fileSizeSameType"
                minOccurs="1"      maxOccurs="1"/>
        </xsd:choice>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="fileMatchConditionType">
    <xsd:sequence>
        <xsd:element name="pattern" type="conditionPatternType"
            minOccurs="0"      default="*.*"/>
        <xsd:element name="exclude" type="conditionPatternType"
            minOccurs="0"      maxOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="fileNoMatchConditionType">
    <xsd:sequence>
        <xsd:element name="pattern" type="conditionPatternType"
            minOccurs="0"      default="*.*"/>
        <xsd:element name="exclude" type="conditionPatternType"
            minOccurs="0"      maxOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="fileSizeConditionType">
    <xsd:sequence>
        <xsd:element name="compare" type="sizeCompareType"
            minOccurs="1"      default="0"/>
        <xsd:element name="pattern" type="conditionPatternType"
            minOccurs="0"      default="*.*"/>
        <xsd:element name="exclude" type="conditionPatternType"
            minOccurs="0"      maxOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="sizeCompareType">
    <xsd:simpleContent>

```

```

        <xsd:extension base="xsd:int">
        <xsd:attribute name="operator" type="sizeOperatorType"
use="required"/>
        <xsd:attribute name="units" type="fileSizeUnitsType"
use="required"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:simpleType name="sizeOperatorType">
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="="/>
    </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="fileSizeUnitsType">
    <xsd:restriction base="xsd:string">
        <xsd:pattern value="[bB] | [kK] [bB] | [mM] [bB] | [gG] [bB]" />
    </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="conditionPatternType">
    <xsd:simpleContent>
        <xsd:extension base="xsd:string">
            <xsd:attribute name="type" type="patternTypeAttributeType"
use="optional" default="wildcard" />
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:simpleType name="patternTypeAttributeType">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="regex" />
        <xsd:enumeration value="wildcard" />
    </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="conditionNameType">
    <xsd:simpleContent>
        <xsd:extension base="xsd:string" />
    </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="queueNotEmptyConditionType" />

<xsd:complexType name="completeGroupsConditionType" />

<xsd:complexType name="fileSizeSameType">
    <xsd:sequence>
        <xsd:element name="pattern" type="conditionPatternType"
minOccurs="1" maxOccurs="1" />
        <xsd:element name="exclude" type="conditionPatternType"
minOccurs="0" maxOccurs="1" />
    </xsd:sequence>
    <xsd:attribute name="polls" type="positiveIntegerType" use="required" />
</xsd:complexType>

<xsd:complexType name="pollIntervalType">
    <xsd:simpleContent>
        <xsd:extension base="xsd:int">
            <xsd:attribute name="units" type="timeUnitsType"
use="optional" default="minutes" />
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="batchType">
    <xsd:attribute name="maxSize" type="positiveIntegerType" use="required" />
</xsd:complexType>

<xsd:simpleType name="timeUnitsType">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="seconds" />
        <xsd:enumeration value="minutes" />
        <xsd:enumeration value="hours" />
        <xsd:enumeration value="days" />
        <xsd:enumeration value="weeks" />
        <xsd:enumeration value="months" />
        <xsd:enumeration value="years" />
    </xsd:restriction>
</xsd:simpleType>

```

```

<xsd:complexType name="monitorTasksType">
  <xsd:sequence>
    <xsd:element name="task" type="monitorTaskType"
      minOccurs="1" maxOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="monitorTaskType">
  <xsd:sequence>
    <xsd:element name="name" type="monitorTaskNameType"
      minOccurs="1" maxOccurs="1"/>
    <xsd:element name="description" type="xsd:string"
      minOccurs="0" maxOccurs="1"/>
    <xsd:element name="transfer" type="transferTaskType"
      minOccurs="0" maxOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="transferTaskType">
  <xsd:sequence>
    <xsd:element name="request" type="transferRequestType"
      minOccurs="1" maxOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="resourceIdType">
  <xsd:attribute name="id" type="xsd:string" use="optional"/>
</xsd:complexType>

<xsd:simpleType name="resourceIdAttrType">
  <xsd:restriction base="xsd:string"></xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="monitorNameType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[^%\*]+"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="agentNameType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[_%0-9A-Z]*"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="monitorTaskNameType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value=".*"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="defaultVariablesType">
  <xsd:sequence>
    <xsd:element name="variable" type="variableType"
      maxOccurs="unbounded" minOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="variableType">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute name="key" type="xsd:string" use="required"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>

</xsd:schema>

```

Description du message de création de moniteur

Les éléments et attributs utilisés dans les messages de création de moniteur sont décrits ci-dessous:

Descriptions des éléments

<monitor>

Élément de groupe contenant tous les éléments requis pour annuler un transfert de fichier en cours.

Attribut	Description
version	Indique la version de cet élément fournie par Managed File Transfer.

< nom >

Nom du moniteur, unique dans l'agent du moniteur.

<description >

Description du moniteur (non utilisé actuellement).

<pollInterval >

Intervalle de temps entre chaque vérification de la ressource par rapport à la condition de déclenchement.

Attribut	Description
unités	Indique les unités de temps de l'intervalle d'interrogation. Les valeurs admises sont : <ul style="list-style-type: none"> • secondes • minutes • heures • jours • semaines • mois • années

< agent >

Nom de l'agent auquel le moniteur est associé.

<Resources >

Élément de groupe qui contient les éléments spécifiant les ressources à surveiller.

<répertoire >

Chemin qualifié complet spécifiant le répertoire sur la machine de l'agent du moniteur à surveiller.

Attribut	Description
recursionLevel	Nombre de sous-répertoires à surveiller en plus du répertoire spécifié.
ID	Identificateur unique de la ressource.

< file_attente >

Nom de la file d'attente spécifiant la file d'attente à surveiller sur le gestionnaire de files d'attente de l'agent de surveillance.

<triggerMatch >

Élément de groupe qui contient les éléments spécifiant les conditions de déclenchement à comparer à la ressource contrôlée.

<conditions >

Élément de groupe qui contient les éléments spécifiant le type de condition à comparer à la ressource contrôlée.

<allOf >

Prédicat qui indique que toutes les conditions contenues doivent être satisfaites.

<anyOf >

Prédicat qui indique que toutes les conditions contenues doivent être satisfaites.

< condition >

Définit une condition de comparaison qui contribuera à la condition de déclenchement globale du moniteur.

< nom >

Nom de la condition.

<resource >

Identifie la définition de ressource à laquelle la condition doit être comparée.

Attribut	Description
ID	Identificateur unique de la ressource.

Si la ressource surveillée est un répertoire, l'un des trois éléments suivants doit être spécifié dans la condition:

- fileMatch
- fileNocorrespondance
- fileSize

Si la ressource surveillée est une file d'attente, l'un des deux éléments suivants doit être spécifié dans la condition:

- queueNotvide
- completeGroups

<fileMatch >

Élément de groupe pour une condition de correspondance de nom de fichier.

< modèle >

Indique un modèle de correspondance de nom de fichier. Les fichiers de la ressource doivent correspondre au modèle afin de satisfaire à la condition. Le modèle par défaut est * (tout fichier correspond).

<fileNoCorrespondance >

Élément de groupe pour une condition de correspondance de nom de fichier inverse.

< modèle >

Indique un modèle de correspondance de nom de fichier inverse. Si aucun fichier de la ressource contrôlée ne correspond, la condition est satisfaite. Le modèle par défaut est * (l'absence de fichier correspond).

<fileSize >

Élément de groupe pour une comparaison de taille de fichier.

< compare >

Indique une comparaison de taille de fichier. Cette valeur doit être un nombre entier non négatif.

Attribut	Description
opérateur	Opérateur de comparaison à utiliser. Seul > = est pris en charge.
unités	Indique les unités de taille de fichier, qui peuvent être l'une des suivantes: <ul style="list-style-type: none"> • B-octets • Ko-kilooctets • Mo-mégaoctets • Giga-Go La valeur des unités est insensible à la casse, donc mb'fonctionne aussi bien que MB'.

< modèle >

Modèle de nom de fichier à mettre en correspondance. La valeur par défaut est * (tout fichier correspond).

<queueNotvide >

Cela ne peut être spécifié que si la ressource est une file d'attente. Indique qu'un message doit figurer dans la file d'attente pour que le moniteur soit déclenché.

<completeGroups>

Cela ne peut être spécifié que si la ressource est une file d'attente. Indique qu'un groupe complet de messages doit être présent dans la file d'attente pour que le moniteur soit déclenché. Une tâche de transfert unique est exécutée pour chaque groupe complet de la file d'attente.

< réponse>

Élément facultatif utilisé pour spécifier la file d'attente de réponses pour les demandes asynchrones.

Attribut	Description
QMGR	Nom du gestionnaire de files d'attente

< tâches>

Élément de groupe devant contenir des éléments qui spécifient les tâches à appeler lorsque les conditions de déclenchement du moniteur sont satisfaites.

< tâche>

Élément de groupe qui définit une tâche individuelle que le moniteur appelle lorsque les conditions de déclenchement sont satisfaites. Actuellement, une seule tâche peut être spécifiée.

< nom>

Nom de la tâche. Accepte tous les caractères alphanumériques.

<description>

Description de la tâche. Toute valeur de texte est autorisée.

< transfer>

Élément de groupe qui définit une tâche de transfert.

<request>

Élément de groupe qui définit le type de tâche. Il doit contenir l'un des éléments suivants qui sont hérités de la définition de schéma `FileTransfer.xsd` :

- [managedTransfer](#)
- `managedCall`

Attribut	Description
version	Version de la demande fournie par Managed File Transfer. Il s'agit de la forme n.mm où n est la version d'édition principale et mm la version secondaire. Par exemple, 1.00.

< originaire>

Élément de groupe qui contient les éléments spécifiant l'émetteur de la demande.

<hostName>

Nom d'hôte du système sur lequel se trouve le fichier source.

<userID>

ID utilisateur à l'origine du transfert de fichiers.

< IDmqmdUser>

Facultatif. ID utilisateur IBM MQ fourni dans le descripteur de message (MQMD).

< travail>

Élément de groupe contenant des informations sur le travail.

<jobName>

Indique l'identificateur de travail logique.

<defaultVariables>

Élément de groupe contenant un ou plusieurs éléments variable . Ces variables sont utilisées dans la substitution de variables lors de la surveillance d'une file d'attente. Pour plus d'informations sur la

substitution de variables, voir «Personnalisation des tâches du moniteur de ressources MFT avec la substitution de variables», à la page 185.

< variable >

Élément contenant la valeur associée à la clé fournie par l'attribut key .

Attribut	Description
clé	Nom de la variable par défaut.

Description du message de suppression du moniteur

Les éléments et attributs utilisés dans les messages du moniteur de suppression sont décrits ci-dessous:

Descriptions des éléments

<deleteMonitor >

Élément de groupe contenant tous les éléments requis pour arrêter et supprimer un moniteur.

Attribut	Description
version	Indique la version de cet élément fournie par Managed File Transfer.

< nom >

Nom du moniteur à supprimer.

< originaire >

Élément de groupe qui contient les éléments spécifiant l'émetteur de la demande.

<hostName >

Nom d'hôte du système sur lequel se trouve le fichier source.

<userID >

ID utilisateur à l'origine du transfert de fichiers.

< IDmqmdUser >

Facultatif. ID utilisateur IBM MQ fourni dans le descripteur de message (MQMD).

< réponse >

Indique le nom de la file d'attente de réponses temporaire générée pour la demande. Le nom de la file d'attente est celui défini par la clé `dynamicQueuePrefix` dans le fichier de configuration `command.properties`. S'il n'est pas spécifié, le nom de la file d'attente a la valeur par défaut `WMQFTE`.

Attribut	Description
QMGR	Nom du gestionnaire de files d'attente de commandes sur lequel la file d'attente dynamique temporaire est générée pour recevoir les réponses.

Exemples

Des exemples de messages XML conformes à ce schéma sont fournis pour chacune des demandes de surveillance suivantes:

- [Créer un moniteur](#)
- [Supprimer un moniteur](#)

Exemples de message de demande de moniteur MFT

Exemples de messages que vous pouvez placer dans la file d'attente de commandes de l'agent pour demander à l'agent de créer ou de supprimer un moniteur.

Demande de création de moniteur

```
<?xml version="1.0" encoding="UTF-8"?>
<monitor:monitor xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xmlns:monitor="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition"
  version="4.00"
  xsi:schemaLocation="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition ./
Monitor.xsd">
  <name>EXAMPLEMONITOR</name>
  <pollInterval>1</pollInterval>
  <agent>US2.BINDINGS.FILE</agent>
  <resources>
    <directory recursionLevel="0">/srv/nfs/incoming</directory>
  </resources>
  <triggerMatch>
    <conditions>
      <allOf>
        <condition>
          <fileMatch>
            <pattern>*.completed</pattern>
          </fileMatch>
        </condition>
      </allOf>
    </conditions>
  </triggerMatch>
  <reply QMGR="US2.BINDINGS">WMQFTE.4D400F8B20003702</reply>
  <tasks>
    <task>
      <name/>
      <transfer>
        <request xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
          version="4.00"
          xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
          <managedTransfer>
            <originator>
              <hostName>example.com.</hostName>
              <userID>mqm</userID>
            </originator>
            <sourceAgent QMgr="US2.BINDINGS" agent="US2.BINDINGS.FILE"/>
            <destinationAgent QMgr="US2.BINDINGS" agent="US2.BINDINGS.FILE"/>
            <transferSet>
              <item checksumMethod="MD5" mode="binary">
                <source disposition="leave" recursive="false">
                  <file>/srv/nfs/incoming/*.txt</file>
                </source>
                <destination exist="error" type="directory">
                  <file>/srv/backup</file>
                </destination>
              </item>
            </transferSet>
          </managedTransfer>
        </request>
      </transfer>
    </task>
  </tasks>
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
  </originator>
</monitor:monitor>
```

Demande de suppression de moniteur

```
<?xml version="1.0" encoding="UTF-8"?>
<monitor:deleteMonitor xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xmlns:monitor="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition"
  version="4.00"
  xsi:schemaLocation="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition ./
Monitor.xsd">
  <name>EXAMPLEMONITOR</name>
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
  </originator>
  <reply QMGR="QM_JUPITER">WMQFTE.4D400F8B20003705</reply>
</monitor:deleteMonitor>
```

Référence associée

«Formats des messages de demande du moniteur MFT», à la page 822

Les moniteurs de ressources sont créés lorsqu'un message XML approprié arrive dans la file d'attente de commandes d'un agent, généralement suite à l'émission par un utilisateur de la commande de surveillance fteCreateou à l'utilisation de l'interface IBM MQ Explorer .

Format de message de demande de l'agent MFT Ping

Vous pouvez envoyer une commande ping à un agent en émettant une commande **ftePingAgent** ou en plaçant un message XML dans la file d'attente de commandes de l'agent. La requête XML de l'agent ping doit être conforme au schéma PingAgent.xsd . Une fois que vous avez installé Managed File Transfer, vous pouvez trouver le fichier schéma PingAgent.xsd dans le répertoire suivant: `MQ_INSTALLATION_PATH/mqft/samples/schema`. Le PingAgent.xsd schéma importe fteutils.xsd, qui se trouve dans le même répertoire.

Lorsque l'agent reçoit un message de demande d'agent ping dans sa file d'attente de commandes, s'il est actif, il renvoie un message de réponse XML à la commande ou à l'application qui place le message de demande d'agent ping dans la file d'attente de commandes. Le message de réponse de l'agent est au format défini par Reply.xsd. Pour plus d'informations sur ce format, voir «Format du message de réponse de l'agent MFT», à la page 832.

Schéma

Le schéma suivant décrit les éléments valides dans un message XML de demande d'agent ping.

```
<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema"
  xmlns="https://www.ibm.com/xmlns/wmqfte/7.0.1/PingAgent"
  targetNamespace="https://www.ibm.com/xmlns/wmqfte/7.0.1/PingAgent">

  <xsd:include schemaLocation="fteutils.xsd"/>

  <xsd:element name="pingAgent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="originator" type="origRequestType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="agent" type="agentType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="reply" type="replyType" maxOccurs="1" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="version" type="versionType" use="required"/>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

Description du message de demande de l'agent ping

Les éléments et les attributs utilisés dans les messages de demande de l'agent ping sont décrits dans la liste suivante:

<pingAgent>

Élément de groupe contenant tous les éléments requis pour spécifier une demande d'agent ping.

<originaire>

Élément de groupe contenant tous les éléments requis pour spécifier l'émetteur de la demande ping.

<hostName>

Nom d'hôte de la machine sur laquelle la demande a été émise.

<userID>

Nom d'utilisateur de l'émetteur de la demande.

<IDmqmdUser>

Nom d'utilisateur MQMD de l'émetteur de la demande.

<agent>

Agent à envoyer une commande PING.

Attribut	Description
Agent	Obligatoire. Nom de l'agent.
QMgr	Facultatif. Gestionnaire de files d'attente auquel l'agent se connecte.

< réponse >

Nom de la file d'attente à laquelle l'agent doit envoyer le message de réponse.

Attribut	Description
QMGR	Obligatoire. Nom du gestionnaire de files d'attente dans lequel se trouve la file d'attente de réponses.

Exemple

Cet exemple illustre un message d'agent ping envoyé à l'agent AGENT_JUPITER. Si AGENT_JUPITER est actif et peut traiter les demandes d'agent, il envoie un message de réponse à la file d'attente WMQFTE.4D400F8B20003708 sur QM_JUPITER.

```
<?xml version="1.0" encoding="UTF-8"?>
<ping:pingAgent xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xmlns:ping="https://www.ibm.com/xmlns/wmqfte/7.0.1/PingAgent"
  version="4.00">
  <originator>
    <hostName>example.com.</hostName>
    <userID>mqm</userID>
  </originator>
  <agent agent="AGENT_JUPITER" QMgr="QM_JUPITER"/>
  <reply QMGR="QM_JUPITER">WMQFTE.4D400F8B20003708</reply>
</ping:pingAgent>
```

Format du message de réponse de l'agent MFT

Lorsqu'un agent reçoit un message XML dans sa file d'attente de commandes d'agent, si une réponse est requise, l'agent envoie un message de réponse XML à la file d'attente de réponses définie dans le message d'origine. Le code XML de réponse est conforme au schéma Reply .xsd . Le document de schéma Reply .xsd se trouve dans le répertoire MQ_INSTALLATION_PATH/mqft/samples/schema . Le Reply .xsd schéma importe fteutils .xsd, qui se trouve dans le même répertoire.

Schéma

Le schéma suivant décrit les éléments valides dans un message XML de réponse.

```
<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema">
  <xsd:include schemaLocation="TransferLog.xsd"/>
  <xsd:element name="reply">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="transferSet" type="transferSetType" minOccurs="0"
maxOccurs="1"/>
        <xsd:element name="status" type="statusType" minOccurs="1" maxOccurs="1"/>
      </xsd:sequence>
      <xsd:attribute name="version" type="versionType" use="required"/>
      <xsd:attribute name="ID" type="IDType" use="required"/>
      <xsd:attribute name="detailedReplyMessagesDisabled" type="xsd:boolean"
use="optional"/>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```


Remarques

:NONE.

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service IBM puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM EMEA Director of Licensing
IBM Corporation
Tour Descartes
Armonk, NY 10504-1785
U.S.A.

Pour toute demande d'informations relatives au jeu de caractères codé sur deux octets, contactez le service de propriété intellectuelle IBM ou envoyez vos questions par courrier à l'adresse suivante :

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japon

Le paragraphe suivant ne s'applique ni au Royaume-Uni, ni dans aucun pays dans lequel il serait contraire aux lois locales. LE PRESENT DOCUMENT EST LIVRE "EN L'ETAT" SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et logiciels décrits dans ce document.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Corporation
Coordinateur d'interopérabilité logicielle, département 49XA
3605 Autoroute 52 N

Rochester, MN 55901
U.S.A.

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans le présent document et tous les éléments sous disponibles s'y rapportant sont fournis par IBM conformément aux dispositions du Contrat sur les produits et services IBM, aux Conditions Internationales d'Utilisation de Logiciels IBM ou de tout autre accord équivalent.

Les données de performance indiquées dans ce document ont été déterminées dans un environnement contrôlé. Par conséquent, les résultats peuvent varier de manière significative selon l'environnement d'exploitation utilisé. Certaines mesures évaluées sur des systèmes en cours de développement ne sont pas garanties sur tous les systèmes disponibles. En outre, elles peuvent résulter d'extrapolations. Les résultats peuvent donc varier. Il incombe aux utilisateurs de ce document de vérifier si ces données sont applicables à leur environnement d'exploitation.

Les informations concernant des produits non IBM ont été obtenues auprès des fournisseurs de ces produits, par l'intermédiaire d'annonces publiques ou via d'autres sources disponibles. IBM n'a pas testé ces produits et ne peut confirmer l'exactitude de leurs performances ni leur compatibilité. Elle ne peut recevoir aucune réclamation concernant des produits non IBM. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Toute instruction relative aux intentions d'IBM pour ses opérations à venir est susceptible d'être modifiée ou annulée sans préavis, et doit être considérée uniquement comme un objectif.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

Licence sur les droits d'auteur :

Le présent logiciel contient des exemples de programmes d'application en langage source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation des plateformes pour lesquels ils ont été écrits ou aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes.

Si vous visualisez ces informations en ligne, il se peut que les photographies et illustrations en couleur n'apparaissent pas à l'écran.

Documentation sur l'interface de programmation

Les informations d'interface de programmation, si elles sont fournies, sont destinées à vous aider à créer un logiciel d'application à utiliser avec ce programme.

Ce manuel contient des informations sur les interfaces de programmation prévues qui permettent au client d'écrire des programmes pour obtenir les services de WebSphere MQ.

Toutefois, lesdites informations peuvent également contenir des données de diagnostic, de modification et d'optimisation. Ces données vous permettent de déboguer votre application.

Important : N'utilisez pas ces informations de diagnostic, de modification et d'optimisation en tant qu'interface de programmation car elles sont susceptibles d'être modifiées.

Marques

IBM, le logo IBM, ibm.com, sont des marques d'IBM Corporation dans de nombreux pays. La liste actualisée de toutes les marques d'IBM est disponible sur la page Web "Copyright and trademark

information"www.ibm.com/legal/copytrade.shtml. Les autres noms de produits et de services peuvent être des marques d'IBM ou d'autres sociétés.

Microsoft et Windows sont des marques de Microsoft Corporation aux Etats-Unis et/ou dans d'autres pays.

UNIX est une marque de The Open Group aux Etats-Unis et dans certains autres pays.

Linux est une marque de Linus Torvalds aux Etats-Unis et/ou dans certains autres pays.

Ce produit inclut des logiciels développés par le projet Eclipse (<http://www.eclipse.org/>).

Java ainsi que tous les logos et toutes les marques incluant Java sont des marques d'Oracle et/ou de ses sociétés affiliées.



Référence :

(1P) P/N: