

9.4

IBM MQ Riferimento alla configurazione

IBM

Nota

Prima di utilizzare queste informazioni e il prodotto che supportano, leggere le informazioni in [“Informazioni particolari” a pagina 269](#).

Questa edizione si applica alla versione 9 release 4 di IBM® MQ e a tutte le successive release e modifiche se non diversamente indicato nelle nuove edizioni.

Quando si inviano informazioni a IBM, si concede a IBM un diritto non esclusivo di utilizzare o distribuire le informazioni in qualsiasi modo ritenga appropriato senza incorrere in alcun obbligo verso l'utente.

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

Indice

Riferimento di configurazione.....	5
Esempio: impostazione della comunicazione tra piattaforme per IBM MQ.....	5
Come utilizzare gli esempi di comunicazione tra piattaforme.....	7
Esempio: impostazione della comunicazione tra piattaforme IBM MQ su AIX.....	9
Esempio: impostazione della comunicazione tra piattaforme IBM MQ su IBM i.....	15
Esempio: impostazione della comunicazione tra piattaforme IBM MQ su Linux.....	32
Esempio: impostazione della comunicazione tra piattaforme IBM MQ su Windows.....	39
Example: setting up IBM MQ cross-platform communication on z/OS.....	45
Example: setting up IBM MQ cross-platform communication on z/OS using QSGs.....	50
Example: setting up IBM MQ cross-platform communication for intra-group queuing on z/OS.....	57
Autorizzazioni del file system IBM MQ applicate a /var/mqm.....	65
Autorizzazioni file IBM MQ in /opt/mqm con setuid per mqm.....	69
Autorizzazioni del file system IBM MQ su Windows.....	71
Limitazioni di denominazione per code.....	72
Limitazioni di denominazione per altri oggetti.....	74
Risoluzione nome coda.....	75
Qual è la risoluzione del nome della coda?.....	77
Come vengono risolti gli attributi degli oggetti di destinazione per gli alias, le code remote e le code cluster?.....	78
Oggetti di sistema e predefiniti.....	78
SYSTEM.BASE.TOPIC.....	83
Informazioni sulla stanza dei file di configurazione.....	84
Stanze del file di configurazione per l'accodamento distribuito.....	87
Attributi canale.....	88
Attributi canale per parole chiave MQSC (A-B).....	93
Attributi di canale per parole chiave MQSC (C).....	97
Attributi del canale per le parole chiave MQSC (D-L).....	104
Attributi del canale per le parole chiave MQSC (M).....	111
Attributi canale per le parole chiave MQSC (N-R).....	116
Attributi canale per le parole chiave MQSC (S).....	121
Attributi del canale per le parole chiave MQSC (T-Z).....	126
Comandi e attributi del cluster IBM MQ.....	128
Attributi cluster disponibili nei comandi di definizioni di canale.....	129
Attributi cluster disponibili sui comandi di definizione della coda.....	132
Attributi cluster disponibili nei comandi di definizione del gestore code.....	134
DISPLAY CLUSQMgr.....	135
Aggiornamento cluster.....	137
RESET CLUSTER: rimozione forzata di un gestore code da un cluster.....	138
SUSPEND QMgr, RESUME QMgr e cluster.....	140
Bilanciamento del carico di lavoro nei cluster.....	141
Asynchronous behavior of CLUSTER commands on z/OS.....	150
Programmi canale.....	151
Lavori di intercomunicazione su IBM i.....	151
Stati del canale su IBM i.....	151
Esempio: pianificazione di un canale di messaggi in AIX, Linux, and Windows.....	152
Impostazione dell'esempio di canale di messaggi per AIX, Linux, and Windows.....	154
Esecuzione ed espansione dell'esempio per AIX, Linux, and Windows.....	155
Esempio: pianificazione di un canale di messaggi in IBM i.....	156
Impostazione dell'agent del canale dei messaggi su IBM i.....	157
Esecuzione ed espansione dell'esempio per IBM i.....	160
Example: planning a message channel on z/OS.....	161
Setting up the message channel agent on z/OS.....	162

Running and expanding the example for z/OS.....	164
Example: planning a message channel for z/OS using queue sharing groups.....	165
Setting up the queue sharing group definitions and a queue manager QM3 not in the queue sharing group.....	167
Running the queue sharing group example for z/OS.....	168
Utilizzo di un alias per fare riferimento a una libreria MQ.....	169
Managed File Transfer Riferimento di configurazione.....	169
L'utilizzo delle variabili di ambiente nelle proprietà MFT.....	169
File MFT installation.properties.....	171
Il file MFT agent.properties.....	175
File MFT coordination.properties.....	199
File MFT command.properties.....	203
File MFT logger.properties.....	207
Output prodotto dalla funzione LogTransfer.....	217
Java proprietà di sistema per MFT.....	220
SHA-2 CipherSpecs e CipherSuites per MFT.....	221
File di configurazione del programma di registrazione file MFT.....	221
La libreria SCSQFCMD.....	229
SYSTEM.FTE FTE.....	230
MFT Impostazioni coda agent.....	232
Code di sistema MFT e argomento di sistema.....	233
Convenzioni di denominazione degli oggetti MFT.....	235
Messaggi di stato dell'agente MFT.....	236
IBM MQ Internet Pass-Thru Riferimento di configurazione.....	237
Riepilogo delle proprietà MQIPT.....	238
MQIPT Proprietà globali.....	243
MQIPT Proprietà instradamento.....	247
Proprietà di mqiptAdmin	267
Informazioni particolari.....	269
Informazioni sull'interfaccia di programmazione.....	270
Marchi.....	270

Riferimento di configurazione

Utilizzare le informazioni di riferimento in questa sezione per configurare IBM MQ.

Le informazioni di riferimento per la configurazione sono fornite nei seguenti topic secondari:

Attività correlate

[Configurazione](#)


 [Configurazione di z/OS](#)

Esempio: impostazione della comunicazione tra piattaforme per IBM MQ

Questo esempio mostra come stabilire una rete IBM MQ funzionante configurando i canali mittente e destinatario IBM MQ per abilitare il flusso di messaggi bidirezionale tra le piattaforme su tutti i protocolli supportati.










Prima di iniziare

Gli esempi di configurazione presuppongono che siano presenti particolari infrastrutture di rete per particolari piattaforme:

-  z/OS comunica utilizzando un controller di rete 3745 (o equivalente)

Si presume inoltre che, per SNA, tutte le definizioni richieste in VTAM e NCP (network control program) siano attive e attivate per le piattaforme collegate alla LAN per comunicare sulla WAN (wide area network). Allo stesso modo, per TCP, si presume che la funzione del server dei nomi sia disponibile, utilizzando un server dei nomi di dominio o utilizzando tabelle gestite localmente (ad esempio, un file host).

Le configurazioni di esempio riguardano i seguenti prodotti software di rete:

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

Per ulteriori informazioni sul software e sui protocolli di comunicazione supportati, consultare [Requisiti di sistema per IBM MQ](#).

Informazioni su questa attività

Questo esempio utilizza i canali mittente e destinatario. Per utilizzare tipi di canale diversi da mittente - destinatario, [DEFINE CHANNEL](#) (definire un nuovo canale).

Figura 1 a pagina 6 è una rappresentazione concettuale di un singolo canale e degli oggetti IBM MQ ad esso associati.

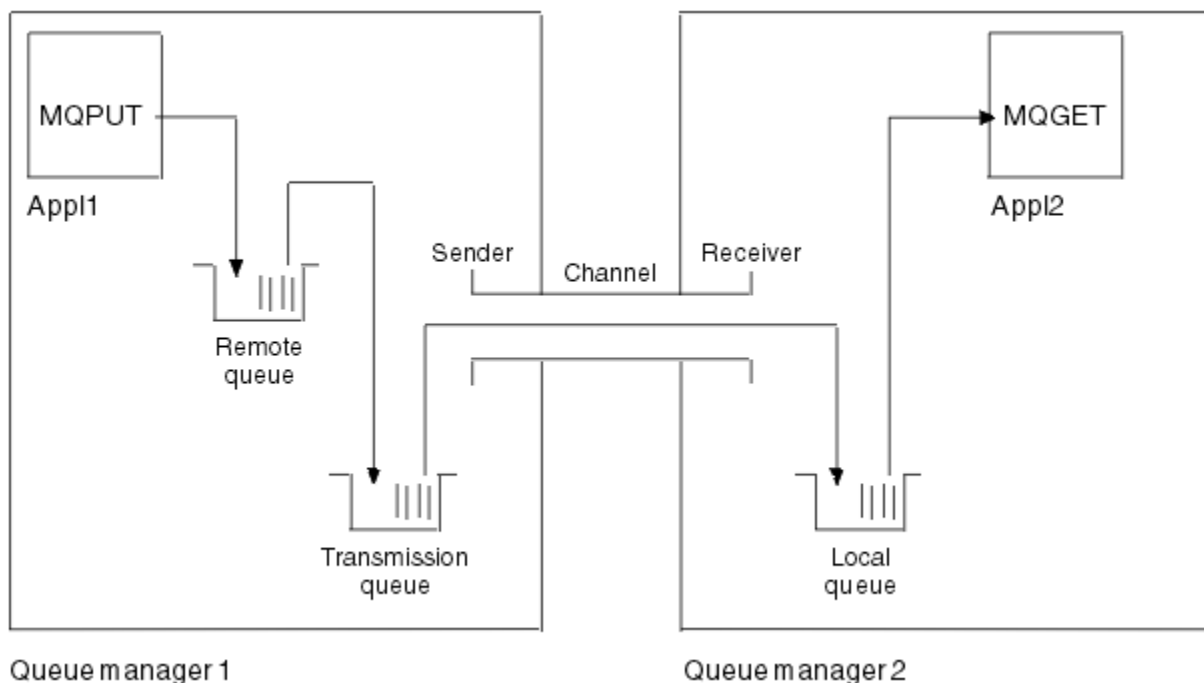


Figura 1. Canale IBM MQ da impostare nella configurazione di esempio

Questo esempio è semplice, per introdurre solo gli elementi di base della rete IBM MQ. Non dimostra l'utilizzo del trigger descritto in [Trigger canali](#).

Gli oggetti in questa rete sono:

- Una coda remota
- Una coda di trasmissione
- Una coda locale
- Un canale mittente
- Un canale ricevente

App1 e App2 sono entrambi programmi applicativi; App1 sta inserendo messaggi e App2 li sta ricevendo.

App1 inserisce i messaggi in una coda remota. La definizione per questa coda remota specifica un gestore code di destinazione, una coda locale su tale gestore code e una coda di trasmissione su questo gestore code locale.

Quando il gestore code riceve la richiesta da App1 di inserire un messaggio nella coda remota, il gestore code determina dalla definizione della coda che la destinazione è remota. Pertanto, inserisce il messaggio, insieme a un'intestazione di trasmissione, direttamente nella coda di trasmissione specificata nella definizione. Il messaggio rimane nella coda di trasmissione fino a quando il canale non diventa disponibile, il che potrebbe verificarsi immediatamente.








Un canale mittente ha nella sua definizione un riferimento a una e una sola coda di trasmissione. Quando un canale viene avviato, e in altre occasioni durante il suo normale funzionamento, esamina questa coda di trasmissione e invia i messaggi su di esso al sistema di destinazione. Il messaggio contiene i dettagli dell'intestazione di trasmissione della coda di destinazione e del gestore code.

Gli esempi di intercomunicazione descrivono in dettaglio la creazione di ciascuno degli oggetti precedenti descritti, per varie combinazioni di piattaforme.

Sul gestore code di destinazione, le definizioni sono richieste per la coda locale e per il lato destinatario del canale. Questi oggetti operano in modo indipendente l'uno dall'altro e quindi possono essere creati in qualsiasi sequenza.

Sul gestore code locale, le definizioni sono richieste per la coda remota, la coda di trasmissione e il lato mittente del canale. Poiché sia la definizione della coda remota che la definizione del canale fanno riferimento al nome della coda di trasmissione, si consiglia di creare prima la coda di trasmissione.

Procedura

1. Leggere le informazioni in [“Come utilizzare gli esempi di comunicazione tra piattaforme”](#) a pagina 7.
2. Seguire le istruzioni per la piattaforma o le piattaforme appropriate per stabilire una connessione di rete e definire i canali.
 - a)  Vedere [“Esempio: impostazione della comunicazione tra piattaforme IBM MQ su AIX”](#) a pagina 9
 - b)  Vedere [“Esempio: impostazione della comunicazione tra piattaforme IBM MQ su IBM i”](#) a pagina 15
 - c)  Vedere [“Esempio: impostazione della comunicazione tra piattaforme IBM MQ su Linux”](#) a pagina 32
 - d)  Vedere [“Esempio: impostazione della comunicazione tra piattaforme IBM MQ su Windows”](#) a pagina 39
 - e)  Vedere [“Example: setting up IBM MQ cross-platform communication on z/OS”](#) a pagina 45
 - f)  Vedere [“Example: setting up IBM MQ cross-platform communication on z/OS using QSGs”](#) a pagina 50
 - g)  Vedere [“Example: setting up IBM MQ cross-platform communication for intra-group queuing on z/OS”](#) a pagina 57

Attività correlate

[Configurazione dell'accodamento distribuito](#)

[Impostazione delle comunicazioni con altri gestori code su z/OS](#)

Come utilizzare gli esempi di comunicazione tra piattaforme

Le configurazioni di esempio per l'impostazione della comunicazione tra piattaforme per IBM MQ descrivono le attività che vengono eseguite su una singola piattaforma per impostare la comunicazione con un'altra piattaforma. Gli esempi descrivono quindi le attività per stabilire un canale di lavoro per tale piattaforma.

Ove possibile, l'intenzione è di rendere le informazioni quanto più generiche possibile. Pertanto, per connettere due gestori code su piattaforme differenti, è necessario fare riferimento solo alle due sezioni pertinenti. Eventuali deviazioni o casi speciali sono evidenziati come tali. È anche possibile connettere due gestori code in esecuzione sulla stessa piattaforma (su macchine diverse o sulla stessa macchina). In questo caso, tutte le informazioni possono essere derivate da una sezione.

ALW Su AIX, Linux, and Windows, prima di cominciare a seguire le istruzioni per la propria piattaforma, è necessario impostare diverse variabili d'ambiente. Eseguire questa operazione immettendo uno dei seguenti comandi:

- ▶ **Linux** ▶ **AIX** Su AIX and Linux:

```
MQ_INSTALLATION_PATH/bin/setmqenv
```

dove `MQ_INSTALLATION_PATH` fa riferimento all'ubicazione in cui è installato IBM MQ . Questo comando imposta le variabili di ambiente per la shell in cui si sta attualmente lavorando. Se si apre un'altra shell, è necessario immettere di nuovo il comando.

- ▶ **Windows** Su Windows:

```
MQ_INSTALLATION_PATH/bin/setmqenv
```

dove `MQ_INSTALLATION_PATH` fa riferimento all'ubicazione in cui è installato IBM MQ .

Esistono esempi in cui è possibile trovare i parametri utilizzati nelle configurazioni di esempio. C'è una breve descrizione di ogni parametro e una guida su dove trovare i valori equivalenti nel sistema. Quando si dispone di una serie di valori propri, assicurarsi di utilizzare tali valori quando si utilizzano gli esempi in questa sezione.

Gli esempi non riguardano la modalità di impostazione delle comunicazioni in cui viene utilizzato il clustering. Per informazioni sull'impostazione delle comunicazioni durante l'uso del cluster, consultare [Configurazione di un cluster di gestore code](#). I valori di configurazione della comunicazione qui forniti sono ancora validi.

Esistono configurazioni di esempio per le piattaforme seguenti:

- ▶ **AIX** [“Esempio: impostazione della comunicazione tra piattaforme IBM MQ su AIX” a pagina 9](#)
- ▶ **IBM i** [“Esempio: impostazione della comunicazione tra piattaforme IBM MQ su IBM i” a pagina 15](#)
- ▶ **Linux** [“Esempio: impostazione della comunicazione tra piattaforme IBM MQ su Linux” a pagina 32](#)
- ▶ **Windows** [“Esempio: impostazione della comunicazione tra piattaforme IBM MQ su Windows” a pagina 39](#)
- ▶ **z/OS** [“Example: setting up IBM MQ cross-platform communication on z/OS” a pagina 45](#)
- ▶ **z/OS** [“Example: setting up IBM MQ cross-platform communication on z/OS using QSGs” a pagina 50](#)
- ▶ **z/OS** [“Example: setting up IBM MQ cross-platform communication for intra-group queuing on z/OS” a pagina 57](#)

Responsabilità IT

Per comprendere la terminologia utilizzata negli esempi, considerare le seguenti istruzioni come punto di partenza.

- Amministratore di sistema: la persona (o gruppo di persone) che installa e configura il software per una piattaforma specifica.
- Amministratore di rete: la persona che controlla la connettività LAN, l'assegnazione di indirizzi LAN, le convenzioni di denominazione di rete e altre attività di rete. Questa persona può trovarsi in un gruppo separato o può far parte del gruppo di amministrazione del sistema.

Nella maggior parte delle installazioni di z/OS , esiste un gruppo responsabile dell'aggiornamento del software ACF/VTAM, ACF/NCPe TCP/IP per il supporto della configurazione di rete. Le persone in questo gruppo sono la principale fonte di informazioni necessarie per la connessione di qualsiasi piattaforma IBM MQ a IBM MQ for z/OS. Possono anche influenzare o imporre le convenzioni di denominazione di rete sulle LAN ed è necessario verificarne l'estensione di controllo prima di creare le definizioni.

- Un tipo specifico di amministratore, ad esempio CICS , è indicato nei casi in cui è possibile descrivere più chiaramente le responsabilità della persona.

Le sezioni di configurazione di esempio non tentano di indicare chi è responsabile e in grado di impostare ciascun parametro. In generale, potrebbero essere coinvolte diverse persone.

Attività correlate

“Esempio: impostazione della comunicazione tra piattaforme per IBM MQ” a pagina 5

Questo esempio mostra come stabilire una rete IBM MQ funzionante configurando i canali mittente e destinatario IBM MQ per abilitare il flusso di messaggi bidirezionale tra le piattaforme su tutti i protocolli supportati.

Riferimenti correlati

[setmqenv](#)

Esempio: impostazione della comunicazione tra piattaforme IBM MQ su AIX

Questo esempio mostra come impostare i link di comunicazione da IBM MQ su AIX a IBM MQ su un'altra piattaforma e stabilire un canale di lavoro per tale piattaforma.

Prima di iniziare

Per informazioni generali su questo esempio e su come utilizzarlo, consultare [“Esempio: impostazione della comunicazione tra piattaforme per IBM MQ” a pagina 5](#) e [“Come utilizzare gli esempi di comunicazione tra piattaforme” a pagina 7](#).

Informazioni su questa attività

Questo esempio riguarda l'impostazione della comunicazione tra piattaforme da IBM MQ su AIX alle seguenti piattaforme:

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

Procedura

1. Stabilire una connessione di rete utilizzando una di queste opzioni.

- Stabilire una connessione LU 6.2 . Per ulteriori informazioni sulla configurazione di SNA su TCP/IP, consultare [Communications Server for AIX Library](#).
- Stabilire una connessione TCP.

Il listener deve essere avviato esplicitamente prima che vengano avviati i canali. Consente ai canali riceventi di avviarsi automaticamente in risposta a una richiesta da un canale di invio in entrata.

Utilizzare il seguente comando per avviare il listener IBM MQ for TCP:

```
runmqclsr -t tcp
```

a. Modificare il file `/etc/services`.

Nota: Per modificare il file `/etc/services`, è necessario essere collegati come superutente o root. Se non si dispone della seguente riga in tale file, aggiungerla come mostrato:

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

- b. Modificare il file `/etc/inetd.conf`. Se non si dispone della seguente riga in tale file, aggiungerla come mostrato, sostituendo `MQ_INSTALLATION_PATH` con la directory di alto livello in cui è installato IBM MQ :

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

- c. Immettere il comando `refresh -s inetd`.

Nota: È necessario aggiungere **root** al gruppo `mqm`. Non è necessario che il gruppo primario sia impostato su `mq`. Finché `mqm` si trova nella serie di gruppi, è possibile utilizzare i comandi. Se si stanno eseguendo solo applicazioni che utilizzano il gestore code, non è necessaria l'autorizzazione del gruppo `mqm`.

2. Una volta stabilita la connessione, definire alcuni canali come descritto in [“Configurazione dei canali su AIX”](#) a pagina 10.

Configurazione dei canali su AIX

Per configurare IBM MQ per la configurazione di esempio su AIX, completare la procedura di configurazione di base per il gestore code, quindi configurare i canali mittente e destinatario.

Prima di iniziare

Nota:

1. Prima di iniziare il processo di installazione, assicurarsi di aver creato l'utente e il gruppo `mqm` e di aver impostato la password.
2. Se l'installazione ha esito negativo a causa di spazio insufficiente nel file system, è possibile aumentare la dimensione nel modo seguente, utilizzando il comando `smit C sna`. (Utilizzare `df` per visualizzare lo stato del filesystem. Ciò indica che il volume logico è pieno.)

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

3. Avviare qualsiasi canale utilizzando il comando:

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

4. I programmi di esempio vengono installati in `MQ_INSTALLATION_PATH/samp`, dove `MQ_INSTALLATION_PATH` rappresenta la directory di alto livello in cui è installato IBM MQ .
5. I log degli errori sono memorizzati in `/var/mqm/qmgrs/qmgrname/errors`.
6. Su AIX, è possibile avviare una traccia dei componenti IBM MQ utilizzando i comandi di traccia IBM MQ standard o utilizzando la traccia di sistema AIX . Consultare [Utilizzo della traccia](#) per ulteriori informazioni sulla traccia IBM MQ e la traccia del sistema AIX .
7. Quando si utilizza l'interprete dei comandi `runmqsc` per immettere comandi di gestione, un segno `+` alla fine di una riga indica che la riga successiva è una continuazione. Assicurarsi che vi sia uno spazio tra l'ultimo parametro e il carattere di continuazione.

Procedura

1. Per eseguire la configurazione di base per il gestore code, completare la seguente procedura:

a) Creare il gestore code dalla riga comandi AIX utilizzando il seguente comando:

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

dove:

aix

È il nome del gestore code

-q

Indica che deve diventare il gestore code predefinito

-u *dlqname*

Specifica il nome della coda messaggi non recapitabile

Questo comando crea un gestore code e una serie di oggetti predefiniti.

b) Avviare il gestore code dalla riga comandi AIX utilizzando il comando:

```
strmqm aix
```

dove *aix* è il nome assegnato al gestore code quando è stato creato.

c) Avviare **runmqsc** dalla riga comandi AIX e utilizzarlo per creare la coda messaggi non distribuibile immettendo il seguente comando:

```
def ql (dlqname)
```

dove *dlqname* è il nome assegnato alla coda messaggi non recapitabile quando è stato creato il gestore code.

2. Configurare i canali per la configurazione di esempio.

Per ulteriori informazioni sui parametri utilizzati nei seguenti esempi, consultare [“Parametri di configurazione del canale per AIX”](#) a pagina 13. In ogni caso, l'esempio mostra il comando MQSC. Avviare **runmqsc** da una riga comandi AIX e immettere ciascun comando oppure creare i comandi in un file di comandi.

Windows Questi esempi sono per la connessione di IBM MQ su AIX con IBM MQ su Windows. Per connettersi a IBM MQ su un'altra piattaforma, utilizzare i valori appropriati dalle tabelle in [“Parametri di configurazione del canale per AIX”](#) a pagina 13 invece dei valori per Windows.

a) Definire il canale mittente come mostrato nei seguenti esempi:

- Utilizzo di SNA:

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

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

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

- Utilizzo di TCP:

```

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

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

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

```

b) Definire il canale ricevente come mostrato nei seguenti esempi:

- Utilizzo di SNA:

```

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

```

- Utilizzo di TCP:

```

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

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

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

```

Nota: Esistono modi alternativi per garantire che i canali riceventi SNA vengano attivati correttamente quando un canale mittente avvia una conversazione.

Durante il processo di configurazione di AIX Communications Server, è stato creato un profilo TPN LU 6.2 , che conteneva il percorso completo di un programma eseguibile TP. Nell'esempio, il file è stato chiamato `u/interops/AIX.crs6a`. È possibile scegliere un nome, ma considerare di includere il nome del proprio gestore code. Il contenuto del file eseguibile deve essere:

```

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

```

dove `aix` è il nome del gestore code (A) e `MQ_INSTALLATION_PATH` è la directory di alto livello in cui è installato IBM MQ . Dopo aver creato questo file, abilitarlo per l'esecuzione eseguendo il comando:

```

chmod 755 /u/interops/AIX.crs6a

```

In alternativa alla creazione di un file eseguibile, è possibile specificare il percorso nel pannello Aggiungi profilo TPN LU 6.2 , utilizzando i parametri della riga comandi.

La specifica di un percorso in uno di questi due modi assicura che i canali riceventi SNA vengano attivati correttamente quando un canale mittente avvia una conversazione.

Parametri di configurazione del canale per AIX

I parametri necessari per configurare i canali per la configurazione di esempio su AIX.

Il passo “2” a pagina 11 di “Configurazione dei canali su AIX” a pagina 10 descrive la configurazione da eseguire sul gestore code di AIX per implementare il canale descritto in “Esempio: impostazione della comunicazione tra piattaforme per IBM MQ” a pagina 5. Gli esempi in “Configurazione dei canali su AIX” a pagina 10 sono per la connessione di IBM MQ for IBM i e IBM MQ for Windows. Per connettersi a IBM MQ su un'altra piattaforma, utilizzare i valori della tabella appropriata al posto dei valori per Windows.

Nota: Le parole in **grassetto** sono valori suggeriti e riflettono i nomi degli oggetti IBM MQ utilizzati in questi esempi. È possibile modificarli nell'installazione del prodotto ma, in caso affermativo, accertarsi di utilizzare i propri valori quando si utilizzano gli esempi in questa sezione

Definizione per il nodo locale

Tabella 1. Esempi di configurazione per la definizione per il nodo locale

ID	Nome del parametro	Riferimenti	Esempio utilizzato
A	Nome gestore code		AIX
B	Nome coda locale		AIX.LOCALQ

Connessione a IBM MQ su Windows

Windows

I valori in questa sezione della tabella devono corrispondere a quelli utilizzati in “Parametri di configurazione del canale per Windows” a pagina 43, come indicato.

Tabella 2. Esempi di configurazione per la connessione a IBM MQ su Windows

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	winnt
D	Nome coda remota		WINNT.REMOTEQ
E	Nome coda nel sistema remoto	B	WINNT.LOCALQ
F	Nome coda di trasmissione		winnt
G	Nome canale mittente (SNA)		AIX.WINNT.SNA
H	Nome canale mittente (TCP/IP)		AIX.WINNT.TCP
I	Nome canale ricevitore (SNA)	G	WINNT.AIX.SNA
J	Nome canale ricevitore (TCP)	H	WINNT.AIX.TCP

Connessione a IBM MQ su Linux

Linux

I valori in questa sezione della tabella devono corrispondere a quelli utilizzati in “Parametri di configurazione del canale per Linux” a pagina 36, come indicato.

Tabella 3. Esempi di configurazione per la connessione a IBM MQ su Linux

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	Linux
D	Nome coda remota		LINUX.REMOTEQ
E	Nome coda nel sistema remoto	B	LINUX.LOCALQ
F	Nome coda di trasmissione		Linux
G	Nome canale mittente (SNA)		AIX.LINUX.SNA
H	Nome canale mittente (TCP/IP)		AIX.LINUX.TCP
I	Nome canale ricevitore (SNA)	G	LINUX.AIX.SNA
J	Nome canale destinatario (TCP/IP)	H	LINUX.AIX.TCP

Connessione a IBM MQ su IBM i

IBM i

I valori in questa sezione della tabella devono corrispondere a quelli utilizzati in “Parametri di configurazione del canale per IBM i” a pagina 29, come indicato.

Tabella 4. Esempi di configurazione per la connessione a IBM MQ su IBM i

ID	Nome parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	AS400
D	Nome coda remota		AS400.REMOTEQ
E	Nome coda nel sistema remoto	B	AS400.LOCALQ
F	Nome coda di trasmissione		AS400
G	Nome canale mittente (SNA)		AIX.AS400.SNA
H	Nome canale mittente (TCP)		AIX.AS400.TCP
I	Nome canale ricevitore (SNA)	G	AS400.AIX.SNA
J	Nome canale ricevitore (TCP)	H	AS400.AIX.TCP

Connessione a IBM MQ for z/OS

z/OS

I valori in questa sezione della tabella devono corrispondere a quelli utilizzati in “Channel configuration parameters for z/OS” a pagina 48, come indicato.

Tabella 5. Esempi di configurazione per la connessione a IBM MQ for z/OS

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	Multiple Virtual Storage (MVS)
D	Nome coda remota		MVS.REMOTEQ
E	Nome coda nel sistema remoto	B	MVS.LOCALQ
F	Nome coda di trasmissione		Multiple Virtual Storage (MVS)

Tabella 5. Esempi di configurazione per la connessione a IBM MQ for z/OS (Continua)

ID	Nome del parametro	Riferimenti	Esempio utilizzato
G	Nome canale mittente (SNA)		AIX.MVS.SNA
H	Nome canale mittente (TCP)		AIX.MVS.TCP
I	Nome canale ricevitore (SNA)	G	MVS.AIX.SNA
J	Nome canale ricevitore (TCP)	H	MVS.AIX.TCP

Connessione a IBM MQ for z/OS utilizzando i gruppi di condivisione code



I valori in questa sezione della tabella devono corrispondere a quelli utilizzati in “[Shared channel configuration parameters](#)” a pagina 56, come indicato.

Tabella 6. Esempi di configurazione per la connessione a IBM MQ for z/OS utilizzando i gruppi di condivisione code

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	QSG
D	Nome coda remota		QSG.REMOTEQ
E	Nome coda nel sistema remoto	B	QSG.SHAREDQ
F	Nome coda di trasmissione		QSG
G	Nome canale mittente (SNA)		AIX.QSG.SNA
H	Nome canale mittente (TCP)		AIX.QSG.TCP
I	Nome canale ricevitore (SNA)	G	QSG.AIX.SNA
J	Nome canale ricevitore (TCP)	H	QSG.AIX.TCP

IBM i Esempio: impostazione della comunicazione tra piattaforme IBM MQ su IBM i

Questo esempio mostra come impostare i link di comunicazione da IBM MQ su IBM i a IBM MQ su un'altra piattaforma e stabilire un canale di lavoro per tale piattaforma.


Prima di iniziare

Per informazioni generali su questo esempio e su come utilizzarlo, consultare “[Esempio: impostazione della comunicazione tra piattaforme per IBM MQ](#)” a pagina 5 e “[Come utilizzare gli esempi di comunicazione tra piattaforme](#)” a pagina 7.

Informazioni su questa attività

Questo esempio riguarda l'impostazione della comunicazione tra piattaforme da IBM MQ su IBM i alle seguenti piattaforme:

- Windows
- AIX
- Linux

-  z/OS o MVS
- VSE/ESA

Procedura

1. Stabilire una connessione di rete utilizzando una di queste opzioni.
 - Stabilire una connessione LU 6.2 come descritto in [“Stabilire una connessione LU 6.2 su IBM i”](#) a pagina 16.
 - Stabilire una connessione TCP come descritto in [“Stabilire una connessione TCP su IBM i”](#) a pagina 25.
2. Una volta stabilita la connessione, definire alcuni canali come descritto in [“Configurazione dei canali su IBM i”](#) a pagina 26.

Stabilire una connessione LU 6.2 su IBM i

Per stabilire una connessione LU 6.2 su IBM i, è necessario configurare il nodo locale e collegarlo a un nodo partner.

Informazioni su questa attività

Per ulteriori informazioni sui parametri necessari per configurare la comunicazione dal sistema IBM i a una delle altre piattaforme IBM MQ, consultare le tabelle in [“Parametri di configurazione per una connessione LU 6.2 su IBM i”](#) a pagina 20. I numeri tra parentesi () nei passi dell'attività corrispondono ai valori nella colonna *ID* di queste tabelle.

Per configurare il nodo locale, è necessario:

- Creazione di una descrizione linea
- Aggiungere una voce di instradamento, quindi avviare il sottosistema

Per connettersi a un nodo partner, è necessario:

- Creare una descrizione unità di controllo
- Creare una descrizione unità
- Creare le informazioni sul lato CPI-C
- Aggiungere una voce di comunicazioni per APPC
- Aggiungere una voce dell'elenco di configurazione

Procedura

1. Configurare il nodo locale creando una descrizione linea e aggiungendo una voce di instradamento.
 - a) Creare una descrizione linea.

Se la descrizione della linea non è già stata creata, utilizzare il comando **CRTLINTRN** per specificare i valori per **Descrizione linea** (6) e **Nome risorsa** (7) come mostrato nel seguente esempio:


```

Create Line Desc (token-ring) (CRTLINTRN)

Type choices, press Enter.

Line description . . . . . TOKENRINGL Name
Resource name . . . . . LIN041 Name, *NWID
NWI type . . . . . *FR *FR, *ATM
Online at IPL . . . . . *YES *YES, *NO
Vary on wait . . . . . *NOWAIT *NOWAIT, 15-180 (1 second)
Maximum controllers . . . . . 40 1-256
Attached NWI . . . . . *NONE Name, *NONE

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

```

b) Aggiungere una voce di instradamento.

Immettere il comando **ADDRTGE** e premere Invio, quindi specificare il proprio valore per **Descrizione sottosistema (5)** e i valori riportati nel seguente esempio per **Numero di sequenza della voce di instradamento, Valore di confronto (8), Posizione iniziale, Programma da richiamare Libreria** contenente il programma da richiamare.

```

Add Routing Entry (ADDRTGE)

Type choices, press Enter.

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

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

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

```

Avviare il sottosistema immettendo il comando STRSBS *subsystem description* (5) e premendo Invio.

2. Creare il collegamento con il nodo partner creando una descrizione unità di controllo, una descrizione unità e le informazioni lato CPI-C e aggiungendo una specifica di comunicazioni per APPC e una voce dell'elenco di configurazione.

Windows Questo esempio è per una connessione a un sistema Windows , ma la procedura è la stessa per gli altri nodi.

a) Creare una descrizione unità di controllo

Su un tipo di riga comandi CRTCTLAPPC e premere Invio, quindi specificare i valori per **Descrizione unità di controllo (12)**, impostare **Tipo collegamento** su *LANE impostare **In linea all'IPL** su *NO.

```

Create Ctl Desc (APPC) (CRTCTLAPPC)

Type choices, press Enter.

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

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

```

Premere Invio due volte, seguito da F10, quindi specificare i valori per **Elenco linee commutate (6)**, **Identificatore rete remota (9)**, **Punto di controllo remoto (10)** e **Indirizzo adattatore remoto LAN (16)** e premere Invio.

```

Create Ctl Desc (APPC) (CRTCTLAPPC)

Type choices, press Enter.

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

```

b) Creare una descrizione unità.

Immettere il comando CRTDEVAPPC e premere Invio, quindi specificare i valori per **Descrizione unità (13)**, **Ubicazione remota (11)**, **Ubicazione locale (3)**, **Identificativo di rete remota (9)** e **Unità di controllo collegata (12)**.

```

Create Device Desc (APPC) (CRTDEVAPPC)

Type choices, press Enter.

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

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

```

Nota: È possibile evitare di dover creare manualmente le descrizioni unità di controllo e unità sfruttando il servizio di configurazione automatica IBM i . Per ulteriori informazioni, consultare la documentazione IBM i.

c) Creare le informazioni lato CPI-C.

Immettere CRTCSI e premere F10, quindi specificare i valori per **Informazioni lato (14)**, **Ubicazione remota (11)**, **Programma di transazione (15)**, **Ubicazione locale (3)**, **Modalità identificativa rete remota (9)** e premere Invio.

```

Create Comm Side Information (CRTCSI)

Type choices, press Enter.

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

Text 'description' . . . . . *BLANK

Additional Parameters

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

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

```

d) Aggiungere una voce di comunicazioni per APPC.

Su una riga comandi, immettere ADDCMNE e premere Invio, quindi specificare i valori per **Descrizione sottosistema (5)** e **Unità (13)** e premere nuovamente Invio.

```

Add Communications Entry (ADDCMNE)

Type choices, press Enter.

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

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

```

e) Aggiungere una voce dell'elenco di configurazione.

Immettere ADDCFGLE *APPNRMT e premere F4, quindi specificare i valori per **Nome ubicazione remota** (11), **Identificativo rete remota** (9), **Nome ubicazione locale** (3), **Punto di controllo remoto** (10) e **ID rete punto di controllo** (9) e premere Invio.

```

Add Configuration List Entries (ADDCFGLE)

Type choices, press Enter.

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

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

```

Operazioni successive

La connessione LU 6.2 è ora stabilita. Si è pronti a completare la configurazione come descritto in [“Configurazione dei canali su IBM i”](#) a pagina 26.

Attività correlate

[“Stabilire una connessione TCP su IBM i”](#) a pagina 25

Se TCP è già configurato, non ci sono attività di configurazione aggiuntive. Se TCP/IP non è configurato, è necessario aggiungere un'interfaccia TCP/IP, aggiungere un'interfaccia di loopback TCP/IP e aggiungere un instradamento predefinito.

Parametri di configurazione per una connessione LU 6.2 su IBM i

I parametri necessari per impostare la comunicazione da IBM MQ su un sistema IBM i a una delle altre piattaforme IBM MQ utilizzando una connessione LU 6.2 .

Utilizzare queste tabelle con le tabelle per la piattaforma a cui si sta effettuando la connessione.

Dove i numeri vengono visualizzati nella colonna *Riferimento*, indicano che il valore deve corrispondere a quello presente nella tabella appropriata altrove in questa sezione. I passi dell'attività in “Stabilire una connessione LU 6.2 su IBM i” a pagina 16 fanno riferimento ai valori nella colonna *ID* di questa tabella.

Le voci nella colonna *Nome parametro* sono illustrate in “Spiegazione dei termini utilizzati nelle tabelle” a pagina 23.

Definizione per il nodo locale

Tabella 7. Esempi di configurazione per la definizione per il nodo locale

ID	Nome del parametro	Riferimenti	Esempio utilizzato
1	ID di rete locale		NETID
2	Nome punto di controllo locale		AS400PU
3	Nome LU		AS400LU
4	Indirizzo di destinazione LAN		10005A5962EF
5	Descrizione del sottosistema		QCMN
6	Descrizione linea		TOKENRINGL
7	Nome risorsa		LIN041
8	Nome programma transazione locale		MQSERIES

Connessione a IBM MQ su Windows

Windows

Windows

Tabella 8. Esempi di configurazione per la connessione a IBM MQ su Windows

ID	Nome del parametro	Riferimenti	Esempio utilizzato
9	ID rete	2	NETID
10	Nome punto di controllo	3	WINNTCP
11	Nome LU	5	WINNTLU
12	Descrizione del controller		WINNTCP
13	Dispositivo		WINNTLU
14	Informazioni laterali		NTCPIC
15	Programma di transazione	7	MQSERIES
16	Indirizzo adattatore LAN	9	08005AA5FAB9
17	Modalità	17	#INTER

Connessione a IBM MQ su AIX

AIX

Tabella 9. Esempi di configurazione per la connessione a IBM MQ sul sistema AIX

ID	Nome del parametro	Riferimenti	Esempio utilizzato
9	ID rete	1	NETID
10	Nome punto di controllo	2	AIXPU

Tabella 9. Esempi di configurazione per la connessione a IBM MQ sul sistema AIX (Continua)

ID	Nome del parametro	Riferimenti	Esempio utilizzato
11	Nome LU	4	AIXLU
12	Descrizione del controller		AIXPU
13	Dispositivo		AIXLU
14	Informazioni laterali		AIXCPIC
15	Programma di transazione	6	MQSERIES
16	Indirizzo adattatore LAN	8	123456789012
17	Modalità	14	#INTER

Connessione a IBM MQ su Linux (piattaformax86)

Linux

Tabella 10. Esempi di configurazioni per la connessione a IBM MQ su Linux (piattaformax86)

ID	Nome del parametro	Riferimenti	Esempio utilizzato
9	ID rete	4	NETID
10	Nome punto di controllo	2	LINUXPU
11	Nome LU	5	LINUXLU
12	Descrizione del controller		LINUXPU
13	Dispositivo		LINUXLU
14	Informazioni laterali		LXCPIC
15	Programma di transazione	7	MQSERIES
16	Indirizzo adattatore LAN	8	08005AC6DF33
17	Modalità	6	#INTER

Connessione a IBM MQ for z/OS

z/OS

Tabella 11. Esempi di configurazione per la connessione a IBM MQ for z/OS

ID	Nome del parametro	Riferimenti	Esempio utilizzato
9	ID rete	2	NETID
10	Nome punto di controllo	3	MVSPU
11	Nome LU	4	MVSLU
12	Descrizione del controller		MVSPU
13	Dispositivo		MVSLU
14	Informazioni laterali		MVSPIC
15	Programma di transazione	7	MQSERIES
16	Indirizzo adattatore LAN	8	400074511092
17	Modalità	6	#INTER

Connessione a un sistema VSE/ESA

ID	Nome parametro	Riferimenti	Esempio utilizzato
9	ID rete	1	NETID
10	Nome punto di controllo	2	VSEPU
11	Nome LU	3	VSELU
12	Descrizione del controller		VSEPU
13	Dispositivo		VSELU
14	Informazioni laterali		VSECPIC
15	Programma di transazione	4	MQ01
16	Indirizzo adattatore LAN	5	400074511092
17	Modalità		#INTER

Spiegazione dei termini utilizzati nelle tabelle

1 2 3

Per informazioni su come trovare i valori configurati, consultare [“Come trovare gli attributi di rete”](#) a pagina 24.

Indirizzo di destinazione 4 LAN

L'indirizzo hardware dell'adattatore token - ring del sistema IBM i . È possibile trovare il valore utilizzando il comando DSPLIND *Line description* (6).

Descrizione sottosistema 5

Questo parametro è il nome di qualsiasi sottosistema IBM i attivo durante l'utilizzo del gestore code. Il nome QCMN è stato utilizzato perché è il sottosistema di comunicazioni IBM i .

6 Descrizione linea

Se questo parametro è stato specificato, viene indicato nel campo Descrizione del nome della risorsa. Per ulteriori informazioni, consultare [“Come trovare il valore del nome della risorsa”](#) a pagina 24. Se il valore non viene specificato, è necessario creare una descrizione di linea.

7 Nome risorsa

Per informazioni su come trovare il valore configurato, consultare [“Come trovare il valore del nome della risorsa”](#) a pagina 24.

8 Nome programma transazione locale

Le applicazioni IBM MQ che tentano di conversare con questa stazione di lavoro specificano un nome simbolico per il programma da eseguire all'estremità ricevente. Questo nome è definito sulla definizione di canale nel mittente. Per semplicità, laddove possibile, utilizzare il nome del programma di transazione MQSERIES oppure, nel caso di una connessione a VSE/ESA, dove la lunghezza è limitata a 4 byte, utilizzare MQTP.

Per ulteriori informazioni, consultare [Impostazioni sul sistema IBM i locale per una piattaforma del gestore code remoto](#) .

Descrizione unità di controllo 12

Questo parametro è un alias per il nome del punto di controllo (o nome nodo) del sistema partner. Per comodità, in questo esempio è stato utilizzato il nome effettivo del partner.

13 Unità

Questo parametro è un alias per la LU del sistema partner. Per comodità, è stato utilizzato il nome LU del partner in questo esempio.

14 Informazioni laterali

Questo parametro è il nome assegnato al profilo di informazioni lato CPI-C. Specificare il proprio nome di 8 caratteri.

Come trovare gli attributi di rete

Il nodo locale è stato configurato parzialmente come parte dell'installazione di IBM i . Per visualizzare gli attributi di rete correnti, immettere il comando **DSPNETA**.

Se è necessario modificare questi valori, utilizzare il comando **CHGNETA**. Per applicare le modifiche potrebbe essere richiesto un IPL.

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

More...
Press Enter to continue.

F3=Exit F12=Cancel
```

Verificare che i valori per **ID rete locale** (1), **Nome punto di controllo locale** (2) e **Ubicazione locale predefinita** (3), corrispondano ai valori nella tabella o ai propri valori, se sono stati modificati.

Come trovare il valore del nome della risorsa

Per trovare il valore del nome risorsa, immettere **WRKHDWRSC TYPE (*CMN)** e premere Invio.

Viene visualizzato il pannello Gestione risorse di comunicazioni. Il valore per **Nome risorsa** viene trovato come porta token - ring. In questo esempio è **LINO41** .


```

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

```

```

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

```

```

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

```

IBM i Stabilire una connessione TCP su IBM i

Se TCP è già configurato, non ci sono attività di configurazione aggiuntive. Se TCP/IP non è configurato, è necessario aggiungere un'interfaccia TCP/IP, aggiungere un'interfaccia di loopback TCP/IP e aggiungere un instradamento predefinito.

Procedura

1. Aggiungere un'interfaccia TCP/IP.

In una riga comandi, immettere ADDTCPIFC e premere Invio, quindi specificare **Indirizzo IP** e **Descrizione linea**, e una **Maschera di sottorete** della macchina e premere nuovamente Invio.

```

Add TCP/IP Interface (ADDTCPIFC)

Type choices, press Enter.

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

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

```

2. Aggiungere un'interfaccia di loopback TCP/IP.

Dalla riga comandi, immettere ADDTCPIFC e premere Invio, quindi specificare i valori per **Indirizzo IP**, **Descrizione linea** e **Maschera di sottorete**.

```

Add TCP Interface (ADDTCPIFC)

Type choices, press Enter.

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

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

```

3. Aggiungere un instradamento predefinito.

Sulla riga comandi, immettere ADDTCP RTE e premere Invio, quindi specificare i valori appropriati per la rete e premere Invio per creare una voce di instradamento predefinita.

```

Add TCP Route (ADDTCP RTE)

Type choices, press Enter.

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

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

```

Operazioni successive

La connessione TCP è ora stabilita. Si è pronti a completare la configurazione come descritto in [“Configurazione dei canali su IBM i”](#) a pagina 26.

Attività correlate

[“Stabilire una connessione LU 6.2 su IBM i”](#) a pagina 16

Per stabilire una connessione LU 6.2 su IBM i, è necessario configurare il nodo locale e collegarlo a un nodo partner.

Configurazione dei canali su IBM i

Per configurare IBM MQ per la configurazione di esempio su IBM i, completare la procedura di configurazione di base per il gestore code, quindi configurare i canali mittente e destinatario.

Informazioni su questa attività

Utilizzare il comando **WRKMQMQ** per visualizzare il menu di configurazione IBM MQ .

Avviare il listener del canale TCP utilizzando il comando **STRMQMLSR**.

Avviare qualsiasi canale mittente utilizzando il comando STRMQMCHL CHLNAME(*channel_name*).

Nota: Gli errori AMQ* vengono inseriti nel log relativo al lavoro che ha rilevato l'errore. Utilizzare il comando **WRKACTJOB** per visualizzare un elenco di lavori. Nel nome del sottosistema QSYSWRK, individuare il lavoro ed immettere 5 su di esso per gestire tale lavoro. I log IBM MQ hanno il prefisso AMQ.

Procedura

1. Creare il gestore code.
 - a) Immettere CRTMQM e premere Invio.

```
                Create Message Queue Manager (CRTMQM)

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

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

Bottom
```

- b) Nel campo **Nome gestore code messaggi** , immettere AS400 e nel campo **Coda messaggi non recapitati** , immettere DEAD.LETTER.QUEUE e premere Invio.
 - c) Avviare il gestore code immettendo STRMQM MQMNAME(AS400).
 - d) Creare la coda messaggi non recapitata utilizzando i seguenti parametri:

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

Per ulteriori informazioni e per un esempio di come definire una coda, consultare il passo [“2” a pagina 27](#).

2. Definire una coda.
 - a) Immettere CRTMQMQ sulla riga comandi.

```

Create MQM Queue (CRTMQMQ)
Type choices, press Enter.
Queue name . . . . .
Queue type . . . . .          *ALS, *LCL, *RMT

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

```

b) Completare i due campi di questo pannello e premere Invio.

Viene quindi visualizzato un altro pannello, con campi di immissione per gli altri parametri di cui si dispone. I valori predefiniti possono essere utilizzati per tutti gli altri attributi della coda.

3. Definire un canale.

a) Immettere CRTMQMCHL sulla riga comandi per visualizzare il riquadro **Crea canale MQM** .

```

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

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


```

b) Completare i due campi di questo pannello e premere Invio.

Viene visualizzato un altro pannello in cui è possibile specificare i valori per gli altri parametri forniti in precedenza. I valori predefiniti possono essere utilizzati per tutti gli altri attributi del canale.

4. Configurare i canali per la configurazione di esempio.

Per ulteriori informazioni sui parametri utilizzati nei seguenti esempi, consultare [“Parametri di configurazione del canale per IBM i”](#) a pagina 29.

 Questi esempi sono per la connessione di IBM MQ su IBM i con IBM MQ su Windows. Per connettersi a IBM MQ su un'altra piattaforma, utilizzare i valori appropriati dalle tabelle in [“Parametri di configurazione del canale per IBM i”](#) a pagina 29 invece dei valori per Windows.

a) Definire il canale mittente come mostrato nei seguenti esempi:

- Utilizzo di SNA:

```

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

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

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

- Utilizzo di TCP:

```

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

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

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

b) Definire il canale ricevente come mostrato nei seguenti esempi:

- Utilizzo di SNA:

```

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

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

- Utilizzo di TCP:

```

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

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

Parametri di configurazione del canale per IBM i

I parametri necessari per configurare i canali per la configurazione di esempio su IBM i.

Il passo “4” a pagina 28 di “Configurazione dei canali su IBM i” a pagina 26 descrive la configurazione da eseguire sul gestore code di IBM i per implementare il canale descritto in “Esempio: impostazione della comunicazione tra piattaforme per IBM MQ” a pagina 5. Gli esempi in “Configurazione dei canali su IBM i” a pagina 26 sono per la connessione di IBM MQ for IBM i e IBM MQ for Windows. Per connettersi a IBM MQ su un'altra piattaforma, utilizzare i valori della tabella appropriata al posto dei valori per Windows.

Nota:

1. Le parole in **grassetto** sono valori suggeriti e riflettono i nomi degli oggetti IBM MQ utilizzati in questi esempi. È possibile modificarli nell'installazione del prodotto, ma, in tal caso, accertarsi di utilizzare i propri valori quando si utilizzano gli esempi riportati in questa sezione.
2. Il comando IBM MQ channel ping (**PNGMQMCHL**) viene eseguito in modo interattivo, mentre l'avvio di un canale causa l'inoltro di un lavoro batch. Se un ping del canale viene completato correttamente, ma il canale non viene avviato, la rete e le definizioni IBM MQ sono probabilmente corrette, ma l'ambiente IBM i per il lavoro batch non lo è. Ad esempio, assicurarsi che QSYS2 venga incluso nella parte di sistema dell'elenco librerie e non solo nell'elenco librerie personale.

Per ulteriori informazioni ed esempi su come creare gli oggetti elencati nelle tabelle, consultare “Configurazione dei canali su IBM i” a pagina 26.

Definizione per il nodo locale

Tabella 13. Esempi di configurazione per la definizione per il nodo locale

ID	Nome del parametro	Riferimenti	Esempio utilizzato
A	Nome gestore code		AS400
B	Nome coda locale		AS400.LOCALQ

Connessione a IBM MQ su Windows



I valori in questa sezione della tabella devono corrispondere ai valori utilizzati in “Parametri di configurazione del canale per Windows” a pagina 43, come indicato.

Tabella 14. Esempi di configurazione per la connessione a IBM MQ su Windows

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	winnt
D	Nome coda remota		WINNT.REMOTEQ
E	Nome coda nel sistema remoto	B	WINNT.LOCALQ
F	Nome coda di trasmissione		winnt
G	Nome canale mittente (SNA)		AS400.WINNT.SNA
H	Nome canale mittente (TCP/IP)		AS400.WINNT.TCP
I	Nome canale ricevitore (SNA)	G	WINNT.AS400.SNA
J	Nome canale destinatario (TCP/IP)	H	WINNT.AS400.TCP

Connessione a IBM MQ su AIX



I valori in questa sezione della tabella devono corrispondere ai valori utilizzati in “Parametri di configurazione del canale per AIX” a pagina 13, come indicato.

Tabella 15. Esempi di configurazione per la connessione a IBM MQ su AIX

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	AIX
D	Nome coda remota		AIX.REMOTEQ
E	Nome coda nel sistema remoto	B	AIX.LOCALQ
F	Nome coda di trasmissione		AIX
G	Nome canale mittente (SNA)		AS400.AIX.SNA
H	Nome canale mittente (TCP/IP)		AS400.AIX.TCP
I	Nome canale ricevitore (SNA)	G	AIX.AS400.SNA
J	Nome canale ricevitore (TCP)	H	AIX.AS400.TCP

Connessione a IBM MQ su Linux

Linux

I valori in questa sezione della tabella devono corrispondere ai valori utilizzati in “Parametri di configurazione del canale per Linux” a pagina 36, come indicato.

Tabella 16. Esempi di configurazione per la connessione a IBM MQ su Linux

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	Linux
D	Nome coda remota		LINUX.REMOTEQ
E	Nome coda nel sistema remoto	B	LINUX.LOCALQ
F	Nome coda di trasmissione		Linux
G	Nome canale mittente (SNA)		AS400.LINUX.SNA
H	Nome canale mittente (TCP/IP)		AS400.LINUX.TCP
I	Nome canale ricevitore (SNA)	G	LINUX.AS400.SNA
J	Nome canale destinatario (TCP/IP)	H	LINUX.AS400.TCP

Connessione a IBM MQ for z/OS

z/OS

I valori in questa sezione della tabella devono corrispondere ai valori utilizzati in “Channel configuration parameters for z/OS” a pagina 48, come indicato.

Tabella 17. Esempi di configurazione per la connessione a IBM MQ for z/OS

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	Multiple Virtual Storage (MVS)
D	Nome coda remota		MVS.REMOTEQ
E	Nome coda nel sistema remoto	B	MVS.LOCALQ
F	Nome coda di trasmissione		Multiple Virtual Storage (MVS)
G	Nome canale mittente (SNA)		AS400.MVS.SNA

Tabella 17. Esempi di configurazione per la connessione a IBM MQ for z/OS (Continua)

ID	Nome del parametro	Riferimenti	Esempio utilizzato
H	Nome canale mittente (TCP)		AS400.MVS.TCP
I	Nome canale ricevitore (SNA)	G	MVS.AS400.SNA
J	Nome canale ricevitore (TCP)	H	MVS.AS400.TCP

Connessione a un sistema VSE/ESA

I valori in questa sezione della tabella devono corrispondere ai valori utilizzati nel sistema VSE/ESA .

Tabella 18. Esempi di configurazione per la connessione a un sistema VSE/ESA

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	VSE
D	Nome coda remota		VSE.REMOTEQ
E	Nome coda nel sistema remoto	B	VSE.LOCALQ
F	Nome coda di trasmissione		VSE
G	Nome canale mittente		AS400.VSE.SNA
I	Nome canale ricevente	G	VSE.AS400.SNA

Linux Esempio: impostazione della comunicazione tra piattaforme IBM MQ su Linux

Questo esempio mostra come impostare i link di comunicazione da IBM MQ su Linux a IBM MQ su un'altra piattaforma e stabilire un canale di lavoro per tale piattaforma.

Prima di iniziare

Per informazioni generali su questo esempio e su come utilizzarlo, consultare [“Esempio: impostazione della comunicazione tra piattaforme per IBM MQ”](#) a pagina 5 e [“Come utilizzare gli esempi di comunicazione tra piattaforme”](#) a pagina 7.

Informazioni su questa attività

Questo esempio riguarda l'impostazione della comunicazione tra piattaforme da IBM MQ su Linux alle seguenti piattaforme:

-  Windows
-  AIX
-  IBM i
-  z/OS

`MQ_INSTALLATION_PATH` rappresenta la directory di alto livello in cui è installato IBM MQ .

Stabilire una connessione di rete utilizzando la LU 6.2 o TCP.

Nota: Per TCP, alcune distribuzioni Linux utilizzano ora il daemon inet esteso (XINETD) invece del daemon inet (INETD). Le seguenti istruzioni indicano come stabilire una connessione TCP utilizzando il daemon inet o il daemon inet esteso.

Procedura

1. Stabilire una connessione di rete utilizzando LU6.2

Nota: Le informazioni in questa sezione si applicano solo a IBM MQ per Linux (piattaforma x86). Non si applica a IBM MQ per la piattaforma Linux (x86-64), IBM MQ per Linux (piattaforma zSeries s390x) o IBM MQ per Linux (piattaforma Power).

Per le informazioni più recenti sulla configurazione di SNA su TCP/IP, fare riferimento al manuale Administration Guide per la propria versione di Linux dalla seguente documentazione: [Communications Server for Data Center Deployment on Linux library](#).

2. Stabilire una connessione TCP utilizzando il daemon inet (INETD)

a) Modificare il file /etc/services.

Se non si dispone della seguente riga nel file, aggiungerla come mostrato:

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

Nota: Per modificare questo file, è necessario essere collegati come superutente o root.

b) Modificare il file /etc/inetd.conf.

Se non si dispone della seguente riga in tale file, aggiungerla come mostrato:

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

c) Trovare l'ID processo di inetd con il comando:

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

d) Esegui il comando:

```
kill -1 inetd processid
```

Se si dispone di più di un gestore code sul sistema e quindi si richiede più di un servizio, è necessario aggiungere una riga per ciascun gestore code aggiuntivo sia a /etc/services che a inetd.conf.

Ad esempio:

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

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

Ciò evita la generazione di messaggi di errore se esiste una limitazione sul numero di richieste di connessione in sospeso accodate su una singola porta TCP. Per informazioni sul numero di richieste di connessioni in sospeso, consultare [Utilizzo dell'opzione di backlog del listener TCP](#).

Il processo inetd su Linux può limitare la frequenza delle connessioni in entrata su una porta TCP. Il valore predefinito è 40 connessioni in un intervallo di 60 secondi. Se è necessaria una velocità maggiore, specificare un nuovo limite sul numero di connessioni in entrata in un intervallo di 60 secondi accodando un punto (.) seguito dal nuovo limite al parametro nowait del servizio appropriato in inetd.conf. Ad esempio, per un limite di 500 connessioni in un intervallo di 60 secondi utilizzare:

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

`MQ_INSTALLATION_PATH` rappresenta la directory di alto livello in cui è installato IBM MQ .

3. Stabilire una connessione TCP utilizzando il daemon inet esteso (XINETD)

Le seguenti istruzioni descrivono come il daemon inet esteso viene implementato su Red Hat Linux. Se si sta utilizzando una distribuzione Linux diversa, potrebbe essere necessario adattare queste istruzioni.

a) Modificare il file `/etc/services`.

Se non si dispone della seguente riga nel file, aggiungerla come mostrato:

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

Se non si dispone della seguente riga nel file, aggiungerla come mostrato:

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

b) Creare un file denominato IBM MQ nella directory di configurazione XINETD, `/etc/xinetd.d` aggiungendo la stanza seguente al file:

```
# IBM MQ service for XINETD
service MQSeries
{
    disable          = no
    flags            = REUSE
    socket_type      = stream
    wait            = no
    user            = mqm
    server          = MQ_INSTALLATION_PATH/bin/amqcsta
    server_args     = -m queue.manager.name
    log_on_failure += USERID
}
```

c) Riavviare il daemon inet esteso emettendo il seguente comando:

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

Se si dispone di più di un gestore code sul sistema e quindi si richiede più di un servizio, è necessario aggiungere una riga a `/etc/services` per ogni gestore code aggiuntivo. È possibile creare un file nella directory `/etc/xinetd.d` per ogni servizio oppure aggiungere ulteriori stanze al file IBM MQ creato in precedenza.

Il processo `xinetd` su Linux può limitare la quantità di connessioni in entrata su una porta TCP. Il valore predefinito è 50 connessioni in un intervallo di 10 secondi. Se è necessaria una velocità più elevata, specificare un nuovo limite per la velocità delle connessioni in entrata specificando l'attributo 'cps' nel file di configurazione `xinetd`. Ad esempio, per un limite di 500 connessioni in un intervallo di 60 secondi utilizzare:

```
cps = 500 60
```

4. Completare la configurazione ora che è stata stabilita la connessione TCP/IP.

Andare a [“Configurazione dei canali su Linux”](#) a pagina 34.

Linux Configurazione dei canali su Linux

Per configurare IBM MQ per la configurazione di esempio su Linux, completare la procedura di configurazione di base per il gestore code, quindi configurare i canali mittente e destinatario.

Prima di iniziare

Prima di iniziare il processo, accertarsi di aver creato l'ID utente `mqm` e il gruppo `mqm` e impostare la password.

Avviare qualsiasi canale utilizzando il comando:

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

Informazioni su questa attività

Note:

1. I programmi di esempio vengono installati in `MQ_INSTALLATION_PATH/samp`, dove `MQ_INSTALLATION_PATH` rappresenta la directory di alto livello in cui è installato IBM MQ .
2. I log degli errori sono memorizzati in `errori /var/mqm/qmgrs/ qmgrname /`.
3. Quando si utilizza l'interprete dei comandi **runmqsc** per immettere i comandi di amministrazione, un segno + alla fine di una riga indica che la riga successiva è una continuazione. Assicurarsi che vi sia uno spazio tra l'ultimo parametro e il carattere di continuazione.

Procedura

1. Impostare la configurazione di base:

- a) Creare il gestore code e una serie di oggetti predefiniti, dal prompt UNIX , utilizzando il comando:

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

dove:

linux

È il nome del gestore code

-q

Indica che deve diventare il gestore code predefinito

-u dlqname

Specifica il nome della DLQ (dead letter queue)

- b) Avviare il gestore code, dal prompt UNIX , utilizzando il comando:

```
strmqm linux
```

dove *linux* è il nome assegnato al gestore code quando è stato creato.

2. Configurare i canali per la configurazione di esempio.

Per ulteriori informazioni sui parametri utilizzati nei seguenti esempi, consultare [“Parametri di configurazione del canale per Linux”](#) a pagina 36. In ogni caso, l'esempio mostra il comando MQSC. Avviare **runmqsc** da una riga comandi Linux e immettere ciascun comando oppure creare i comandi in un file di comandi.

Windows Questi esempi sono per la connessione di IBM MQ su Linux con IBM MQ su Windows. Per connettersi a IBM MQ su un'altra piattaforma, utilizzare i valori appropriati dalle tabelle in [“Parametri di configurazione del canale per Linux”](#) a pagina 36 invece dei valori per Windows.

- a) Definire il canale mittente come mostrato nei seguenti esempi:

- Utilizzo di SNA

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

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

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

- Utilizzo di TCP

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

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

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

b) Definizione del canale ricevente come mostrato nei seguenti esempi:

- Utilizzo di SNA:

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

- Utilizzo di TCP:

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

Linux Parametri di configurazione del canale per Linux

I parametri necessari per configurare i canali per la configurazione di esempio su Linux.

Il passo “2” a pagina 35 di “Configurazione dei canali su Linux” a pagina 34 descrive la configurazione da eseguire sul gestore code di Linux per implementare il canale descritto in “Esempio: impostazione della comunicazione tra piattaforme per IBM MQ” a pagina 5. Gli esempi in “Configurazione dei canali su Linux” a pagina 34 sono per la connessione di IBM MQ for IBM i e IBM MQ for Windows. Per connettersi a IBM MQ su un'altra piattaforma, utilizzare i valori della tabella appropriata al posto dei valori per Windows.

Nota: Le parole in **grassetto** sono valori suggeriti e riflettono i nomi degli oggetti IBM MQ utilizzati in questi esempi. È possibile modificarli nell'installazione del prodotto ma, in caso affermativo, accertarsi di utilizzare i propri valori quando si utilizzano gli esempi in questa sezione

Definizione per il nodo locale

Tabella 19. Esempi di configurazione per la definizione per il nodo locale			
ID	Nome del parametro	Riferimenti	Esempio utilizzato
A	Nome gestore code		Linux
B	Nome coda locale		LINUX.LOCALQ

Connessione a IBM MQ su Windows

Windows

I valori in questa sezione della tabella devono corrispondere a quelli utilizzati in “Parametri di configurazione del canale per Windows” a pagina 43, come indicato.

Tabella 20. Esempi di configurazione per la connessione a IBM MQ su Windows

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	winnt
D	Nome coda remota		WINNT.REMOTEQ
E	Nome coda nel sistema remoto	B	WINNT.LOCALQ
F	Nome coda di trasmissione		winnt
G	Nome canale mittente (SNA)		LINUX.WINNT.SNA
H	Nome canale mittente (TCP/IP)		LINUX.WINNT.TCP
I	Nome canale ricevitore (SNA)	G	WINNT.LINUX.SNA
J	Nome canale ricevitore (TCP)	H	WINNT.LINUX.TCP

Connessione a IBM MQ su AIX

AIX

I valori in questa sezione della tabella devono corrispondere a quelli utilizzati in “Configurazione dei canali su AIX” a pagina 10, come indicato.

Tabella 21. Esempi di configurazione per la connessione a IBM MQ su AIX

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	AIX
D	Nome coda remota		AIX.REMOTEQ
E	Nome coda nel sistema remoto	B	AIX.LOCALQ
F	Nome coda di trasmissione		AIX
G	Nome canale mittente (SNA)		.LINUX.AIX.SNA
H	Nome canale mittente (TCP/IP)		LINUX.AIX.TCP
I	Nome canale ricevitore (SNA)	G	AIX.LINUX.SNA
J	Nome canale destinatario (TCP/IP)	H	AIX.LINUX.TCP

Connessione a IBM MQ for IBM i

IBM i

I valori in questa sezione della tabella devono corrispondere a quelli utilizzati in “Parametri di configurazione del canale per IBM i” a pagina 29, come indicato.

Tabella 22. Esempi di configurazione per la connessione a IBM MQ su IBM i

ID	Nome parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	AS400
D	Nome coda remota		AS400.REMOTEQ
E	Nome coda nel sistema remoto	B	AS400.LOCALQ
F	Nome coda di trasmissione		AS400
G	Nome canale mittente (SNA)		LINUX.AS400.SNA
H	Nome canale mittente (TCP)		LINUX.AS400.TCP
I	Nome canale ricevitore (SNA)	G	AS400.LINUX.SNA
J	Nome canale ricevitore (TCP)	H	AS400.LINUX.TCP

Connessione a IBM MQ for z/OS



I valori in questa sezione della tabella devono corrispondere a quelli utilizzati in “Channel configuration parameters for z/OS” a pagina 48, come indicato.

Tabella 23. Esempi di configurazione per la connessione a IBM MQ for z/OS

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	Multiple Virtual Storage (MVS)
D	Nome coda remota		MVS.REMOTEQ
E	Nome coda nel sistema remoto	B	MVS.LOCALQ
F	Nome coda di trasmissione		Multiple Virtual Storage (MVS)
G	Nome canale mittente (SNA)		LINUX.MVS.SNA
H	Nome canale mittente (TCP)		LINUX.MVS.TCP
I	Nome canale ricevitore (SNA)	G	MVS.LINUX.SNA
J	Nome canale ricevitore (TCP)	H	MVS.LINUX.TCP

Connessione a IBM MQ for z/OS utilizzando i gruppi di condivisione code



I valori in questa sezione della tabella devono corrispondere a quelli utilizzati in “Shared channel configuration parameters” a pagina 56, come indicato.

Tabella 24. Esempi di configurazione per la connessione a IBM MQ for z/OS utilizzando i gruppi di condivisione code

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	QSG
D	Nome coda remota		QSG.REMOTEQ
E	Nome coda nel sistema remoto	B	QSG.SHAREDQ

Tabella 24. Esempi di configurazione per la connessione a IBM MQ for z/OS utilizzando i gruppi di condivisione code (Continua)

ID	Nome del parametro	Riferimenti	Esempio utilizzato
F	Nome coda di trasmissione		QSG
G	Nome canale mittente (SNA)		LINUX.QSG.SNA
H	Nome canale mittente (TCP)		LINUX.QSG.TCP
I	Nome canale ricevitore (SNA)	G	QSG.LINUX.SNA
J	Nome canale ricevitore (TCP)	H	QSG.LINUX.TCP

Windows Esempio: impostazione della comunicazione tra piattaforme IBM MQ su Windows

Questo esempio mostra come impostare i link di comunicazione da IBM MQ su Windows a IBM MQ su un'altra piattaforma e stabilire un canale di lavoro per tale piattaforma.

Prima di iniziare

Per informazioni generali su questo esempio e su come utilizzarlo, consultare [“Esempio: impostazione della comunicazione tra piattaforme per IBM MQ”](#) a pagina 5 e [“Come utilizzare gli esempi di comunicazione tra piattaforme”](#) a pagina 7.

Informazioni su questa attività

Questo esempio riguarda l'impostazione della comunicazione tra piattaforme da IBM MQ su Windows alle seguenti piattaforme:

-  AIX
-  IBM i
-  Linux
-  z/OS

Procedura

1. Stabilire una connessione di rete utilizzando LU6.2.

Consultare [AnyNet SNA su TCP/IP e Communications Server for Windows](#) per informazioni sulla configurazione di AnyNet SNA su TCP/IP.

2. Stabilire una connessione di rete utilizzando TCP.

Lo stack TCP fornito con i sistemi Windows non include un daemon *inet* o equivalente.

È necessario avviare esplicitamente il listener prima di avviare i canali. Consente ai canali riceventi di avviarsi automaticamente in risposta a una richiesta da un canale di invio in entrata.

Utilizzare il seguente comando per avviare il listener TCP IBM MQ :

```
runmqclsr -t tcp
```

3. Stabilire una connessione di rete utilizzando NetBIOS.

- a) Ad ogni estremità del canale, specifica il nome NetBIOS locale che deve essere utilizzato dai processi del canale IBM MQ nel file di configurazione del gestore code `qm.ini`.

Ad esempio, la stanza NETBIOS in Windows all'estremità di invio potrebbe essere simile alla seguente:

```
NETBIOS:  
LocalName=WNTNETB1
```

e all'estremità ricevente, è simile al seguente:

```
NETBIOS:  
LocalName=WNTNETB2
```

Ogni processo IBM MQ deve utilizzare un nome NetBIOS locale diverso. non utilizzare il nome del sistema come nome NetBIOS poiché Windows lo utilizza già.

- b) Ad ogni estremità del canale, verificare il numero dell'adattatore LAN utilizzato sul sistema.

Il valore predefinito IBM MQ for Windows per l'adattatore logico numero 0 è NetBIOS in esecuzione su una rete Internet Protocol . Per utilizzare NetBIOS nativo, è necessario selezionare l'adattatore logico numero 1. Consultare [Come stabilire il numero dell'adattatore LAN](#).

Specificare il corretto numero dell'adattatore LAN nella sezione NETBIOS del registro di Windows .
Ad esempio:

```
NETBIOS:  
AdapterNum=1
```

- c) In modo che l'avvio del canale mittente funzioni, specificare il nome NetBIOS locale mediante la variabile di ambiente MQNAME:

```
SET MQNAME=WNTNETB1I
```

Nota: Questo nome deve essere univoco.

- d) All'estremità di invio, definire un canale specificando il nome NetBIOS utilizzato all'altra estremità del canale.

Ad esempio:

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

È necessario specificare l'opzione MCATYPE (THREAD) in quanto, su Windows, i canali mittente devono essere eseguiti come thread.

- e) All'estremità ricevente, definire il canale ricevente corrispondente.

Ad esempio:

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

- f) Avviare l'iniziatore di canali.

Ogni nuovo canale viene avviato come un thread piuttosto che come un nuovo processo:

```
runmqchi
```

- g) Alla fine della ricezione, avviare il listener IBM MQ :


```
runmqclsr -t netbios
```

Facoltativamente è possibile specificare i valori per il nome del gestore code, il nome locale NetBIOS , il numero di sessioni, il numero di nomi e il numero di comandi. Per ulteriori informazioni sull'impostazione di connessioni NetBIOS , consultare [Definizione di una connessione NetBIOS su Windows](#) .

4. Completare la configurazione una volta stabilita la connessione di rete. Consultare [“Configurazione dei canali su Windows”](#) a pagina 41.

Windows Configurazione dei canali su Windows

Per configurare IBM MQ per la configurazione di esempio su Windows, completare la procedura di configurazione di base per il gestore code, quindi configurare i canali mittente e destinatario.

Informazioni su questa attività

Note:

1. È possibile utilizzare il programma di esempio, AMQSBCG, per mostrare il contenuto e le intestazioni di tutti i messaggi in una coda. Ad esempio:

```
AMQSBCG q_name qmgr_name
```

mostra il contenuto della coda *q_name* definita nel gestore code *qmgr_name*.

In alternativa, è possibile utilizzare il browser dei messaggi in Esplora risorse di IBM MQ .

2. È possibile avviare qualsiasi canale dal prompt dei comandi utilizzando il comando

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

3. I log degli errori si possono trovare nelle directory *MQ_INSTALLATION_PATH\qmgrs\qmgrname\errors* e *MQ_INSTALLATION_PATH\qmgrs\@system\errors*. In entrambi i casi, i messaggi più recenti si trovano alla fine di *amqerr01.log*.

MQ_INSTALLATION_PATH rappresenta la directory di livello superiore in cui è installato IBM MQ .

4. Quando si utilizza l'interprete dei comandi **runmqsc** per immettere i comandi di amministrazione, un segno + alla fine di una riga indica che la riga successiva è una continuazione. Assicurarsi che vi sia uno spazio tra l'ultimo parametro e il carattere di continuazione.

Procedura

1. Per impostare la configurazione di base utilizzando il prompt dei comandi, completare la seguente procedura:
 - a) Creare il Gestore code e una serie di oggetti predefiniti utilizzando il comando:

```
ctmqm -u dlqname -q winnt
```

dove:

winnt

È il nome del gestore code

-q

Indica che deve diventare il gestore code predefinito

-u dlqname

Specifica il nome della coda messaggi non recapitabile

- b) Avviare il gestore code utilizzando il comando:

```
stimqm winnt
```

dove *winnt* è il nome assegnato al gestore code al momento della creazione.

2. Configurare i canali per la configurazione di esempio.

Per ulteriori informazioni sui parametri utilizzati nei seguenti esempi, consultare [“Parametri di configurazione del canale per Windows”](#) a pagina 43. In ogni caso, l'esempio mostra il comando MQSC. Avviare **runmqsc** da una riga comandi Linux e immettere ciascun comando oppure creare i comandi in un file di comandi. Questi esempi sono per la connessione di IBM MQ for Windows e IBM MQ for AIX. Per connettersi a IBM MQ su un'altra piattaforma, utilizzare i valori appropriati dalle tabelle in [“Parametri di configurazione del canale per Windows”](#) a pagina 43 invece dei valori per IBM MQ for AIX.

a) Definire il canale mittente come mostrato nei seguenti esempi:

• Utilizzo di SNA

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

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

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

• Utilizzo di TCP

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

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

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

b) Definire il canale ricevente come mostrato nei seguenti esempi:

• Utilizzo di SNA:

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

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

• Utilizzo di TCP:

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

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

Operazioni successive

Avvio automatico

IBM MQ for Windows consente di automatizzare l'avvio di un gestore code e del relativo iniziatore di canali, canali, listener e server dei comandi.

Utilizzare lo snap-in Servizi IBM MQ per definire i servizi per il gestore code. Una volta completata correttamente la verifica della configurazione delle comunicazioni, impostare i servizi pertinenti su **automatico** all'interno dello snap-in. Questo file può essere letto dal servizio IBM MQ fornito all'avvio del sistema.

Per ulteriori informazioni, vedere [Amministrazione di IBM MQ](#).

Esecuzione di canali come processi o thread

IBM MQ for Windows fornisce la flessibilità per eseguire canali di invio come processi Windows o thread Windows. È specificato nel parametro MCATYPE nella definizione del canale mittente.

La maggior parte delle installazioni esegue i propri canali di invio come thread, poiché la memoria virtuale e reale richiesta per supportare molte connessioni di canali simultanei è ridotta. Tuttavia, una connessione NetBIOS necessita di un processo separato per l'MCA (Message Channel Agent) di invio.

Parametri di configurazione del canale per Windows

I parametri necessari per configurare i canali per la configurazione di esempio su Windows.

Il passo “2” a pagina 42 di “Configurazione dei canali su Windows” a pagina 41 descrive la configurazione da eseguire sul gestore code di Linux per implementare il canale descritto in “Esempio: impostazione della comunicazione tra piattaforme per IBM MQ” a pagina 5. Gli esempi in “Configurazione dei canali su Windows” a pagina 41 sono per la connessione di IBM MQ for Windows e IBM MQ for AIX. Per connettersi a IBM MQ su un'altra piattaforma, utilizzare i valori della tabella appropriata al posto dei valori per Windows.

Nota: Le parole in **grassetto** sono valori suggeriti e riflettono i nomi degli oggetti IBM MQ utilizzati in questi esempi. È possibile modificarli nell'installazione del prodotto, ma, in tal caso, accertarsi di utilizzare i propri valori quando si utilizzano gli esempi riportati in questa sezione.

In ogni caso viene visualizzato il comando MQSC. Avviare **runmqsc** da un prompt dei comandi e immettere ciascun comando oppure creare i comandi in un file di comandi.

Vengono forniti esempi per la connessione di IBM MQ for Windows e IBM MQ for AIX. Per connettersi a IBM MQ su un'altra piattaforma, utilizzare la serie appropriata di valori dalla tabella al posto di quelli per Windows.

Definizione per il nodo locale

ID	Nome del parametro	Riferimenti	Esempio utilizzato
A	Nome gestore code		winnt
B	Nome coda locale		WINNT.LOCALQ

Connessione a IBM MQ su AIX



I valori in questa sezione della tabella devono corrispondere a quelli utilizzati in “Parametri di configurazione del canale per AIX” a pagina 13, come indicato.

Tabella 26. Esempi di configurazione per la connessione a IBM MQ su AIX

	Nome parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	AIX
D	Nome coda remota		AIX.REMOTEQ
E	Nome coda nel sistema remoto	B	AIX.LOCALQ
F	Nome coda di trasmissione		AIX
G	Nome canale mittente (SNA)		WINNT.AIX.SNA
H	Nome canale mittente (TCP)		WINNT.AIX.TCP
I	Nome canale ricevitore (SNA)	G	AIX.WINNT.SNA
J	Nome canale ricevitore (TCP)	H	AIX.WINNT.TCP

Connessione a IBM MQ su IBM i



I valori in questa sezione della tabella devono corrispondere a quelli utilizzati in “Parametri di configurazione del canale per IBM i” a pagina 29, come indicato.

Tabella 27. Esempi di configurazione per la connessione a IBM MQ su IBM i

ID	Nome parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	AS400
D	Nome coda remota		AS400.REMOTEQ
E	Nome coda nel sistema remoto	B	AS400.LOCALQ
F	Nome coda di trasmissione		AS400
G	Nome canale mittente (SNA)		WINNT.AS400.SNA
H	Nome canale mittente (TCP)		WINNT.AS400.TCP
I	Nome canale ricevitore (SNA)	G	AS400.WINNT.SNA
J	Nome canale ricevitore (TCP)	H	AS400.WINNT.TCP

Connessione a IBM MQ for z/OS



I valori in questa sezione della tabella devono corrispondere a quelli utilizzati in “Channel configuration parameters for z/OS” a pagina 48, come indicato.

Tabella 28. Esempi di configurazione per la connessione a IBM MQ for z/OS

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	Multiple Virtual Storage (MVS)
D	Nome coda remota		MVS.REMOTEQ
E	Nome coda nel sistema remoto	B	MVS.LOCALQ
F	Nome coda di trasmissione		Multiple Virtual Storage (MVS)

Tabella 28. Esempi di configurazione per la connessione a IBM MQ for z/OS (Continua)

ID	Nome del parametro	Riferimenti	Esempio utilizzato
G	Nome canale mittente (SNA)		WINNT.MVS.SNA
H	Nome canale mittente (TCP)		WINNT.MVS.TCP
I	Nome canale ricevitore (SNA)	G	MVS.WINNT.SNA
J	Nome canale ricevitore (TCP)	H	MVS.WINNT.TCP

Connessione a IBM MQ for z/OS utilizzando i gruppi di condivisione code

z/OS

I valori in questa sezione della tabella devono corrispondere a quelli utilizzati in “[Shared channel configuration parameters](#)” a pagina 56, come indicato.

Tabella 29. Esempi di configurazione per la connessione a IBM MQ for z/OS utilizzando i gruppi di condivisione code

ID	Nome del parametro	Riferimenti	Esempio utilizzato
C	Nome gestore code remoto	A	QSG
D	Nome coda remota		QSG.REMOTEQ
E	Nome coda nel sistema remoto	B	QSG.SHAREDQ
F	Nome coda di trasmissione		QSG
G	Nome canale mittente (SNA)		WINNT.QSG.SNA
H	Nome canale mittente (TCP)		WINNT.QSG.TCP
I	Nome canale ricevitore (SNA)	G	QSG.WINNT.SNA
J	Nome canale ricevitore (TCP)	H	QSG.WINNT.TCP

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

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

Before you begin

For background information about this example and how to use it, see “[Esempio: impostazione della comunicazione tra piattaforme per IBM MQ](#)” on page 5 and “[Come utilizzare gli esempi di comunicazione tra piattaforme](#)” on page 7.

About this task

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

- **Windows** Windows
- **AIX** AIX
- **Linux** Linux

-  IBM i
- VSE/ESA

You can also connect any of the following:

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

Procedure

1. Establish a network connection.

- Establishing an LU 6.2 connection

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

- Establishing a TCP connection

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

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

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

2. Configure the channels.

See “[Configuring the channels on IBM MQ for z/OS](#)” on page 46 for details on how you configure the channels.

Configuring the channels on IBM MQ for z/OS

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

Procedure

1. Start the channel initiator using the command:

```
/cpf START CHINIT 1
```

2. Start an LU 6.2 listener using the command:

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

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

3. Start a TCP listener using the command:

```
/cpf START LSTR
```

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

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

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

4. Configure the channels for the example configuration.

For more information about the parameters used in the following examples, see “Channel configuration parameters for z/OS” on page 48. These examples are for connecting IBM MQ for z/OS and IBM MQ for Windows. To connect to IBM MQ on another platform use the values from the appropriate table in “Channel configuration parameters for z/OS” on page 48 instead of the values for Windows.

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

For LU 6.2:

```

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

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

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

```

For TCP:

```

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

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

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

```

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

For LU 6.2:

```

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

Receiver Channel
  Channel name : WINNT.MVS.SNA
                                     I

```

For TCP:

```

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

```

Channel configuration parameters for z/OS

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

Step “4” on page 47 of “Configuring the channels on IBM MQ for z/OS” on page 46 describes the configuration to be performed on the z/OS queue manager to implement the channel described in “Esempio: impostazione della comunicazione tra piattaforme per IBM MQ” on page 5. The examples in “Configuring the channels on IBM MQ for z/OS” on page 46 are for connecting IBM MQ for z/OS and IBM MQ for Windows. To connect to IBM MQ on another platform use the values from the appropriate table in place of the values for Windows.

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

Definition for local node

Table 30. Configuration examples for the definition for the local node

ID	Parameter Name	Reference	Example Used
A	Queue Manager Name		MVS
B	Local queue name		MVS.LOCALQ

Connection to IBM MQ on Windows

The values in this section of the table must match the values used in “Parametri di configurazione del canale per Windows” on page 43, as indicated.



Table 31. Configuration examples for connecting to IBM MQ on Windows

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

Connection to IBM MQ on AIX

The values in this section of the table must match the values used in “Parametri di configurazione del canale per AIX” on page 13, as indicated.

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

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

Connection to IBM MQ on Linux

 Linux

The values in this section of the table must match the values used in “Parametri di configurazione del canale per Linux” on page 36, as indicated.

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

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

Connection to IBM MQ on IBM i

 IBM i

The values in this section of the table must match the values used in “Parametri di configurazione del canale per IBM i” on page 29, as indicated.

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

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

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

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

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

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

Before you begin

Setting up communication links from a queue sharing group to a platform other than z/OS is the same as described in [“Example: setting up IBM MQ cross-platform communication on z/OS”](#) on page 45.

For background information about this example and how to use it, see [“Esempio: impostazione della comunicazione tra piattaforme per IBM MQ”](#) on page 5 and [“Come utilizzare gli esempi di comunicazione tra piattaforme”](#) on page 7.

Procedure

1. Establish a network connection using one of the following options.
 - Establish an LU 6.2 connection as described in [“Establishing an LU 6.2 connection into a queue sharing group”](#) on page 50.
 - Establish a TCP connection using Sysplex Distributor as described in [“Establishing a TCP connection using Sysplex Distributor”](#) on page 54.
2. Define some channels to complete the configuration after the connection is established.

See [“Configuring shared channels on IBM MQ for z/OS”](#) on page 55 for details of this process.

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

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

About this task

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

Procedure

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

This line must be of the form:

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

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

The line you add must take the form

```
LUADD ACBNAME(mvslu1)
      NOSCHED
      TPDATA(csq.appctp)
      GRNAME(mvsgtr)
```

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

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

c) Start the APPC subsystem with the command:

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

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

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

```
SET APPC=xx
```

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

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

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

e) Activate the major node.

You can do this activation with the command:

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

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

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

The entries you add will look like this example:

```
SIADD  DESTNAME(G1)           11
       MODENAME(#INTER)
       TPNAME(MQSERIES)
       PARTNER_LU(MVSLU1)     9
SIADD  DESTNAME(G2)           12
```

```

MODENAME (#INTER)
TPNAME (MQSERIES)
PARTNER_LU (MVSGR)      10

```

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

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

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

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

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

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

The entry you add looks like this:

```

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

```

What to do next

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

Go to [“Configuring shared channels on IBM MQ for z/OS ”](#) on page 55.

Configuration parameters for an LU 6.2 connection

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

The steps required to set up an LU 6.2 connection are described in [“Establishing an LU 6.2 connection into a queue sharing group”](#) on page 50, with numbered cross-references to the parameters in the example.

Numbers in the Reference column indicate that the value must match that in the appropriate example elsewhere in this section. The examples that follow in this section refer to the values in the ID column. The entries in the Parameter Name column are explained in [“Explanation of terms”](#) on page 53.

Definition for local node using generic resources

<i>Table 35. Configuration examples for the definition for the local node using generic resources</i>			
ID	Parameter name	Reference	Example used
1	Command prefix		/cpf
2	Network ID		NETID
3	Node name		MVSPU
6	Modename		#INTER
7	Local Transaction Program name		MQSERIES
8	LAN destination address		400074511092
9	Local LU name		MVSLU1
10	Generic resource name		MVSGR
11	Symbolic destination		G1

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

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

Connection to IBM MQ on Windows

Windows

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

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

Connection to IBM MQ on AIX

AIX

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

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

Explanation of terms

1 Command prefix

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

2 Network ID

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

3 Node name

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

9 Local LU name

A logical unit (LU) is software that serves as an interface or translator between a transaction program and the network. It manages the exchange of data between transaction programs. The local LU name is the unique VTAM APPLID of this IBM MQ subsystem. Your network administrator can tell you this value.

11 12 13 Symbolic destination

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

6 14 Modename

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

7 15 Transaction Program name

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

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

8 LAN destination address

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

10 Generic resource name

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

16 Partner LU name

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

21 Remote node ID

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

Establishing a TCP connection using Sysplex Distributor

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

Procedure

1. Define a Distributed DVIPA address as follows:

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

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

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

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

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

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

What to do next

Go to [“Configuring shared channels on IBM MQ for z/OS”](#) on page 55.

Configuring shared channels on IBM MQ for z/OS

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

About this task

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

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

Procedure

1. Start the channel initiator using the command:

```
/cpf START CHINIT
```

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

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

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

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

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

4. Configure the channels for the example configuration.

For more information about the parameters used in the following examples, see [“Shared channel configuration parameters”](#) on page 56. These examples are for connecting IBM MQ for z/OS and Windows. To connect to IBM MQ on another platform, use the appropriate values from the tables in [“Shared channel configuration parameters”](#) on page 56 instead of the values for Windows.

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

Using LU 6.2:

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

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

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

Using TCP

```

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

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

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

```

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

Using LU 6.2:

```

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

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

```

Using TCP:

```

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

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

```

Shared channel configuration parameters

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

Step “4” on page 55 of “[Configuring shared channels on IBM MQ for z/OS](#)” on page 55 describes the configuration to be performed on the z/OS queue manager to implement the channel described in “[Esempio: impostazione della comunicazione tra piattaforme per IBM MQ](#)” on page 5. The examples in “[Configuring shared channels on IBM MQ for z/OS](#)” on page 55 are for connecting IBM MQ for z/OS and Windows. To connect to IBM MQ on another platform, use the values from the appropriate table in place of the values for Windows.

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

Definition for local node

Table 38. Configuration examples for the definition for the local node			
ID	Parameter Name	Reference	Example Used
A	Queue Manager Name		QSG

Table 38. Configuration examples for the definition for the local node (continued)

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

Connection to IBM MQ on Windows

Windows

The values in this section of the table must match the values used in “Parametri di configurazione del canale per Windows” on page 43, as indicated.

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

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

Connection to IBM MQ on AIX

AIX

The values in this section of the table must match the values used in “Parametri di configurazione del canale per AIX” on page 13, as indicated.

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

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

z/OS

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

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

About this task

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

Procedure

1. Set up and run Configuration 1.
For more information, see [“Setting up and running configuration 1”](#) on page 58.
2. Set up and run Configuration 2.
For more information, see [“Setting up and running configuration 2”](#) on page 60.
3. Set up and run Configuration 3.
For more information, see [“Setting up and running configuration 3”](#) on page 63.

What to do next

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

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

Setting up and running configuration 1

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

About this task

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

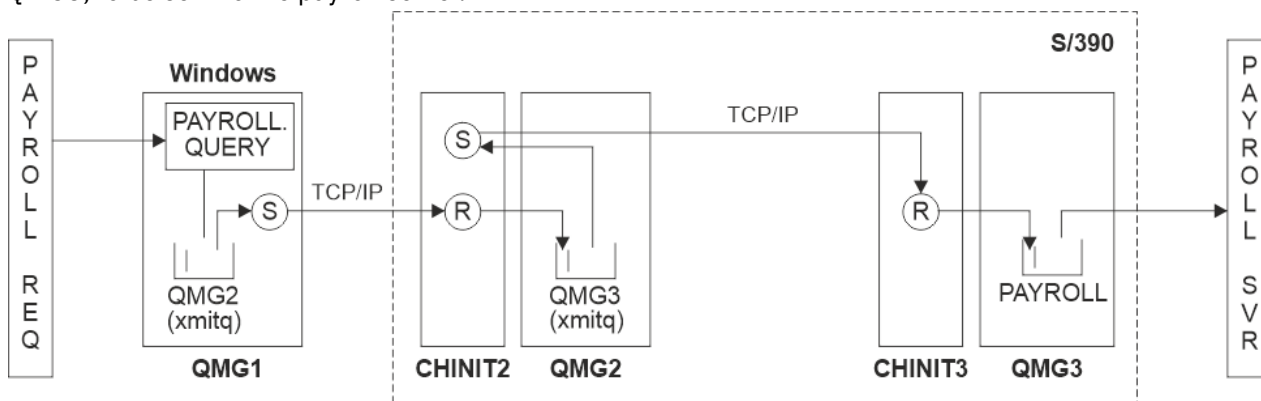


Figure 2. Configuration 1: z/OS using intra-group queuing

The flow of operations is as follows:

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

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

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

Procedure

1. Procedure on QMG1:

- a) Setup the remote queue definition:

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

- b) Setup the transmission queue definition:

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

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

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

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

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

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

- e) Setup the reply-to queue definition:

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

2. Procedure on QMG2:

- a) Setup the transmission queue definition:

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

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

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

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

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

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

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

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

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

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

3. Procedure on QMG3:

a) Setup the local queue definition:

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

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

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

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

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

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

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

4. Use the following procedure to run configuration 1:

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

Setting up and running configuration 2

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

About this task

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

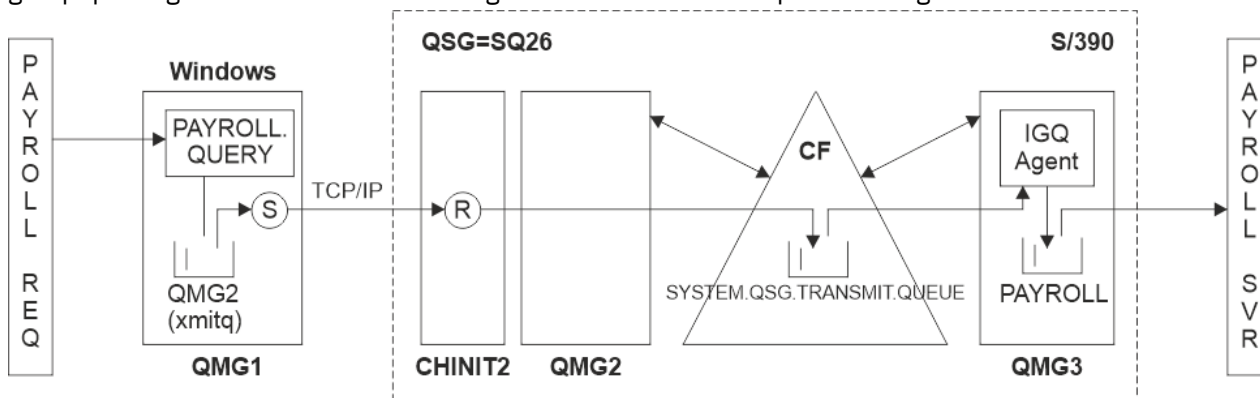


Figure 3. Configuration 2

The flow of operations is as follows:

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

Notes:

- The payroll query example transfers small messages only. If you need to transfer both persistent and non-persistent messages, you can establish a combination of Configuration 1 and Configuration 2, so that large messages can be transferred using the distributed queuing route, while small messages can be transferred using the potentially faster intra-group queuing route.
- The definitions do not take into account triggering, and that only channel definitions for communication using TCP/IP are provided.
- The example assumes that you have already configured queue managers QMG2 and QMG3 to be members of the same queue sharing group.

Procedure

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

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

b) Setup the transmission queue definition:

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

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

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

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

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

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

e) Setup the reply-to queue definition:

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

2. Procedure on QMG2:

a) Setup the transmission queue definition:

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

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

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

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

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

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

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

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

d) Setup the queue manager definition:

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

3. Procedure on QMG3:

a) Setup the local queue definition:

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

b) Setup the queue manager definition:

ALTER QMGR IGQ(ENABLED)

4. Use the following procedure to run configuration 2:
 - a) Start queue managers QMG1, QMG2, and QMG3.
 - b) Start the channel initiator for QMG2.
 - c) Start the listeners on QMG1 to listen on port 1414, and QMG2 to listen on port 1415.
 - d) Start the sender channel on QMG1 and QMG2.
 - e) Start the payroll query requesting application connected to QMG1.
 - f) Start the payroll server application connected to QMG3.
 - g) Submit a payroll query request to QMG3 and wait for the payroll reply.

z/OS Setting up and running configuration 3

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

About this task

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

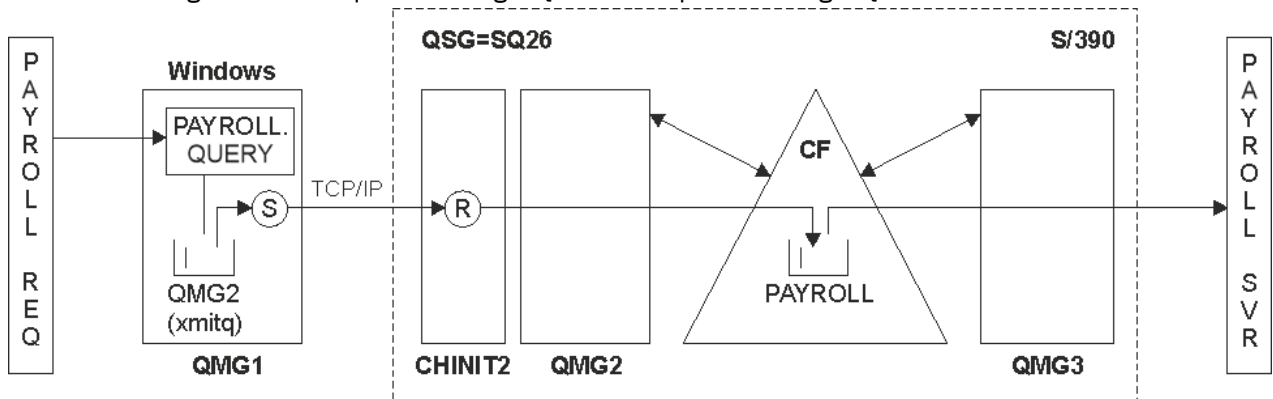


Figure 4. Configuration 3

The flow of operations is:

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

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

For the configuration used to transfer replies back to the payroll request application, see [“Example: planning a message channel for z/OS using queue sharing groups”](#) on page 165.

Notes:

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

Procedure

1. Procedure on QMG1:

a) Setup the remote queue definition:

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

b) Setup the transmission queue definition:

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

c) Setup the sender channel definition:

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

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

d) Setup the transmission channel definition:

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

e) Setup the reply-to queue definition:

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

2. Procedure on QMG2:

a) Setup the transmission queue definition:

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

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

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

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

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

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

d) Setup the local queue definition:

```
DEFINE QLOCAL(PAYROLL) QSGDISP(SHARED) DESCR('Payroll query request queue') +
```



```
REPLACE PUT(ENABLED) USAGE(NORMAL) GET(ENABLED) SHARE +
DEFSOPT(SHARED) DEFPSIST(NO) CFSTRUCT(APPLICATION1)
```

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

3. Use the following procedure to run configuration 3:

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

Linux

IBM i

AIX

Autorizzazioni del file system IBM MQ

applicare a /var/mqm

Le seguenti informazioni descrivono la sicurezza applicata ai file e alle directory in /var/mqm/ e il motivo per cui le autorizzazioni del file system sono impostate così come sono. Per garantire il corretto funzionamento di IBM MQ, non modificare le autorizzazioni del file system come impostato da IBM MQ

crtmqdir comando

Se la propria azienda ha modificato una delle autorizzazioni del file /var/mqm, per qualsiasi motivo, è possibile aggiornare le autorizzazioni o aggiungere directory utilizzando il comando **crtmqdir**

Sicurezza del filesystem IBM MQ su AIX, Linuxe IBM i

I file nella directory di dati IBM MQ (/var/mqm) vengono utilizzati per memorizzare:

- IBM MQ dati di configurazione
- Dati dell'applicazione (oggetti IBM MQ e dati contenuti nei messaggi IBM MQ)
- Informazioni sul controllo di runtime
- Informazioni di monitoraggio (messaggi e file FFST)

L'accesso a questi dati è controllato utilizzando le autorizzazioni del file system con alcuni dei dati accessibili a tutti gli utenti, mentre altri dati sono limitati solo ai membri del IBM MQ gruppo Amministratore 'mqm' (o QMQM su IBM i).

L'accesso è concesso nelle tre categorie seguenti:

solo gruppo mqm

I file e le directory in questa categoria sono accessibili solo agli amministratori IBM MQ (membri del gruppo 'mqm') e ai processi del gestore code IBM MQ.

Le autorizzazioni file per questi file e directory sono:

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

Un esempio di file e directory in questa categoria è:

```
/var/mqm/qmgrs/QMGR/qm.ini
/var/mqm/qmgrs/QMGR/channel/
/var/mqm/qmgrs/QMGR/channel/SYSTEM!DEF!SCRVONN
```

```

/var/mqm/qmgrs/QMGR/queues/
/var/mqm/qmgrs/QMGR/queues/SYSTEM!DEFAULT!LOCAL!QUEUES/
/var/mqm/qmgrs/QMGR/errors/
/var/mqm/qmgrs/QMGR/errors/AMQERR01.LOG
/var/mqm/qmgrs/QMGR/ssl/
/var/mqm/qmgrs/QMGR/@qmgr/
/var/mqm/qmgrs/QMGR/@qmpersist/
...

```

Accesso in lettura a tutti gli utenti - accesso in lettura e scrittura ai membri del gruppo mqm

I file e le directory in questa categoria possono essere letti da tutti gli utenti, ma solo i membri del gruppo mqm possono modificare tali file e manipolare tali directory.

Le autorizzazioni file per questi file e directory sono:

```

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

```

Un esempio di file e directory in questa categoria è:

```

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

```



Attenzione: È necessario impostare solo le autorizzazioni di esecuzione su file e script eseguibili. Ad esempio, su Linux quando il comando **crtmqm** viene eseguito, vengono impostati i seguenti permessi file:

```

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

```

IBM MQ 8.0:

```

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

```

Accesso in lettura e scrittura per tutti gli utenti

File con accesso in lettura e scrittura per tutti gli utenti

IBM MQ non dispone di file *regolari* che dispongono di autorizzazioni file scrivibili world (777). Tuttavia, esistono diversi file *speciali* che sembrano avere autorizzazioni di file scrivibili in tutto il mondo.

Questi file speciali non forniscono alcun rischio per la sicurezza. Sebbene le autorizzazioni siano mostrate come 777, non sono file *regolari* e non è possibile scrivervi direttamente.

Questi file speciali sono:

Collegamenti simbolici

I collegamenti simbolici sono identificati dal carattere 'l' all'inizio delle autorizzazioni. Le autorizzazioni sul collegamento simbolico non hanno alcun effetto su chi è in grado di accedere al file di destinazione, poiché l'accesso al comando è controllato dalle autorizzazioni sulla destinazione del collegamento simbolico.

Nella maggior parte dei sistemi AIX and Linux non è possibile modificare le autorizzazioni sui collegamenti simbolici, quindi appaiono sempre come `lrwxrwxrwx`.

File socket

I file socket sono file speciali creati dal sistema operativo, come risultato di un processo che crea un socket di dominio UNIX . Questi file possono essere identificati da 's' all'inizio delle autorizzazioni file, ovvero `srxwxrwxrwx`.

Le autorizzazioni sul file non concedono l'accesso al file stesso, ma definiscono chi può connettersi al socket del dominio UNIX .

IBM MQ utilizza un numero di questi file socket e le autorizzazioni sono sempre impostate in base a chi è consentito comunicare con il socket.

Le seguenti directory contengono file socket che hanno autorizzazioni di lettura / scrittura per tutti gli utenti (`srxwxrwxrwx`).

IBM MQ 8.0:

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

File socket utilizzati dalle applicazioni che si collegano a IBM MQ utilizzando collegamenti isolati.

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

Directory con accesso in lettura e scrittura per tutti gli utenti

Ci sono momenti in cui le applicazioni IBM MQ hanno bisogno di creare file nella directory di dati IBM MQ . Per garantire che le applicazioni siano in grado di creare file quando sono richiesti, a un numero di directory viene concesso l'accesso in scrittura globale, il che significa che qualsiasi utente sul sistema può creare file all'interno di tale directory.


Ad eccezione dei file di log degli errori, che possono essere scritti da qualsiasi membro del gruppo 'mqm', tutti i file creati in queste directory vengono creati con autorizzazioni limitate che consentono solo l'accesso in scrittura al creatore di file. Ciò consente all'amministratore di sistema di tenere traccia dell'ID utente di tutti i dati scritti nei file in queste directory.

/var/mqm/errors/

Questa directory contiene i file di log degli errori di sistema e i file FFST. L'autorizzazione di questa directory è 'd`srxwxrwsrwt`', il che significa che tutti gli utenti sul sistema possono creare file in questa directory.

Il SetGroupId bit 's' indica che tutti i file creati in questa directory hanno la proprietà del gruppo 'mqm'.

Il bit permanente 't' non è impostato per impostazione predefinita su questa directory, ma un amministratore IBM MQ può impostarla esplicitamente per consentire agli utenti di eliminare solo i file che creano.

Nota:  Questa funzione non è disponibile su IBM i.

AMQERRO*.LOG

Questi file di log degli errori possono essere scritti solo direttamente dai membri di group , ma qualsiasi utente può leggere i messaggi scritti in questi file (autorizzazione: `-rw-rw-r--`).

AMQnnnnn.*.FDC

Questi file contengono le informazioni FFST scritte quando si verifica un errore nel gestore code o in un'applicazione scritta da un utente. Questi file vengono creati con le autorizzazioni `-rw-r-----`.


/var/mqm/trace/

I file di traccia vengono scritti in questa directory quando la traccia IBM MQ è abilitata. La traccia IBM MQ viene scritta da tutti i processi associati a un gestore code per cui la traccia è abilitata.

Le autorizzazioni di questa directory sono 'd`srxwxrwsrwt`', il che significa che tutti gli utenti sul sistema possono creare file in questa directory.

Il bit SetGroupId 's' indica che tutti i file creati in questa directory hanno la proprietà del gruppo 'mqm'.

Il bit permanente 't' non è impostato per impostazione predefinita su questa directory, ma un amministratore IBM MQ può impostarla esplicitamente per consentire agli utenti di eliminare solo i file che creano.

Nota:  Questa funzione non è disponibile su IBM i.

AMQnnnnn.*.TRC

Questi file contengono i dati di traccia scritti da ciascun processo di traccia e creati con le autorizzazioni -rwx-r-----

Le autorizzazioni su questa directory sono drwxrwsrwt e le autorizzazioni dei file socket creati in questa directory sono srwx-----.

IBM MQ 8.0:

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


Questa directory viene utilizzata da applicazioni che si connettono al gestore code IBM MQ utilizzando bind *isolati*. Durante l'elaborazione della connessione, viene creato un file socket dall'applicazione di connessione in questa directory. Il file socket viene rimosso una volta stabilita la connessione al gestore code.

Le autorizzazioni su questa directory sono drwxrwsrwt e le autorizzazioni dei file socket creati in questa directory sono srwx-----.

Il bit SetGroupId 's' su questa directory assicura che tutti i file creati in questa directory abbiano la proprietà del gruppo 'mqm'.

Su tutte le piattaforme tranne IBM i, queste directory hanno anche la serie di bit adesivi 't' che impedisce all'utente di eliminare tutti i file tranne quelli di cui è proprietario. Ciò impedisce a un utente non autorizzato di eliminare i file che non possiede.

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

 Per i processi che si collegano a IBM MQ utilizzando i bind *condivisi*, i socket del dominio UNIX potrebbero essere utilizzati per la sincronizzazione tra l'applicazione e il gestore code. Quando vengono utilizzati i socket del dominio UNIX, il file socket associato viene creato in queste directory.

Le autorizzazioni per queste directory sono drwxrwsrwt e le autorizzazioni per i file socket creati in queste directory sono srwxrwxrwx.

Il bit SetGroupId 's' su queste directory garantisce che tutti i file creati in queste directory abbiano la proprietà del gruppo 'mqm'.

Su tutte le piattaforme tranne IBM i, queste directory hanno anche il bit permanente 't' impostato che impedisce all'utente di eliminare qualsiasi file tranne quelli di cui è proprietario. Ciò impedisce a un utente non autorizzato di eliminare i file che non possiede.

HOME

Una directory `${HOME}/.mqm` viene creata quando si utilizza una versione non registrata o non installata di IBM MQ, ad esempio il client ridistribuibile.

La directory viene creata in modo che IBM MQ abbia un modo affidabile di accedere ai propri file socket utilizzando un percorso che rientra nella lunghezza `sun_path`. Se IBM MQ non è in grado di scrivere nella directory HOME, si riceve un messaggio di errore.

Utilizzo delle risorse System V IPC da parte di IBM MQ

IBM MQ utilizza la memoria condivisa System V e i semafori per la comunicazione tra processi. Queste risorse sono raggruppate in base al modo in cui vengono utilizzate con ciascun gruppo che dispone delle autorizzazioni di accesso e di proprietà appropriate.

Per verificare quale delle risorse System V IPC su un sistema appartiene a IBM MQ, è possibile:

- Controllare la proprietà.

L'utente proprietario delle risorse IBM MQ System V IP è sempre l'utente 'mqm' sulle piattaforme AIX and Linux. Su IBM i l'utente proprietario è 'QMQM'.

- IBM MQ 8.0 e successive, utilizzare il programma di utilità amqspdbg.

Il programma di utilità amqspdbg fornito con IBM MQ può essere utilizzato per visualizzare la memoria condivisa e gli ID semaforo per un determinato gestore code.

È necessario immettere il comando una volta per il gruppo 'system' di risorse System V creato da IBM MQ

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

e quattro volte per ogni gestore code sul sistema per ottenere l'elenco completo di risorse System V utilizzate da IBM MQ. Si supponga un nome gestore code di QMGR1 nei seguenti esempi:

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

Le autorizzazioni di accesso sulle risorse System V create da IBM MQ sono impostate per concedere solo il livello corretto di accesso agli utenti autorizzati. Un certo numero di risorse IPC System V create da IBM MQ sono accessibili a tutti gli utenti sulla macchina e dispongono delle autorizzazioni di -rw-rw-rw-.

Il parametro **-g ApplicationGroup** nel comando `crtmqm` può essere utilizzato per limitare l'accesso a un gestore code all'appartenenza di un determinato gruppo di sistemi operativi. L'utilizzo di questa funzionalità del gruppo limitato limita ulteriormente le autorizzazioni concesse sulle risorse IPC System V.

Linux

AIX

Autorizzazioni file IBM MQ in /opt/mqm con setuid per

mqm

Le seguenti informazioni riguardano la situazione in cui il team di sicurezza ha contrassegnato alcuni dei file eseguibili IBM MQ nella struttura di directory `$MQ_INSTALLATION_PATH`, in violazione delle politiche di sicurezza locali. L'ubicazione predefinita in AIX è `/usr/mqm` e per gli altri sistemi operativi UNIX è `/opt/mqm`. Se IBM MQ è stato installato in una directory non predefinita, come ad esempio `/opt/mqm90`, o se si dispone di più installazioni, i dettagli in questo argomento sono ancora validi.

Causa del problema

Il tuo team di sicurezza ha identificato le seguenti aree di interesse in `$MQ_INSTALLATION_PATH`:

1. I file nella directory `/opt/mqm/bin` sono `setuid` per il proprietario della struttura di directory in cui risiedono. Ad esempio:

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

- Praticamente tutte le directory e i file sono di proprietà di "mqm:mqm" ad eccezione dei seguenti, che sono di proprietà di root:

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

Questa sottodirectory deve essere di proprietà di root, perché questi sono i file eseguibili che interagiscono con il sistema operativo quando l'utente da un client IBM MQ specifica una password e questa password viene passata dal gestore code IBM MQ al sistema operativo per confermare se la password è valida o non è valida.

- L'utente non possiede i file nella /opt/mqm/lib/iconv directory (questa directory non esiste su AIX). Ad esempio:

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

- La directory di manutenzione del fix pack su sistemi Linux basati su RPM. Quando i fix pack sono installati, i file esistenti vengono salvati in questa directory in una struttura simile a quella mostrata nel seguente esempio, ad eccezione del fatto che in questo esempio V.R rappresenta la versione e il numero di release di IBM MQ e le sottodirectory che vengono visualizzate dipendono dai fix pack installati:

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

Risoluzione del problema

Una delle preoccupazioni sui sistemi UNIX rispetto ai programmi setuid era che la sicurezza del sistema potrebbe essere compromessa dalla manipolazione delle variabili di ambiente come LD* (LD_LIBRARY_PATH, LIBPATH su AIX e così via). Questo non è più un problema, poiché diversi sistemi operativi UNIX ora ignorano queste variabili di ambiente LD* durante il caricamento dei programmi setuid.

- Perché alcuni dei programmi IBM MQ sono mqm-setuid o mqm-setgid.

In IBM MQ, l'ID utente "mqm" e qualsiasi ID che fa parte del gruppo "mqm" sono gli utenti amministrativi di IBM MQ.

Le risorse del gestore code IBM MQ vengono protette dall'autenticazione rispetto a questo utente. Poiché i processi del gestore code utilizzano e modificano tali risorse, i processi del gestore code richiedono l'autorizzazione "mqm" per accedere alle risorse. Pertanto, i processi di supporto del gestore code IBM MQ sono progettati per essere eseguiti con l'ID utente effettivo "mqm".

Per consentire agli utenti non amministrativi di accedere agli oggetti IBM MQ, IBM MQ fornisce una funzione OAM (Object Authority Manager), in base alla quale è possibile concedere e revocare le autorizzazioni in base alla necessità dell'applicazione eseguita dall'utente non amministrativo.

Con la possibilità di concedere diversi livelli di autenticazione per gli utenti e il fatto che i programmi **setuid** e **setgid** ignorano le variabili LD*, i file binari e di libreria IBM MQ non compromettono in alcun modo la sicurezza del proprio sistema.

- Non è possibile modificare le autorizzazioni per soddisfare la politica di sicurezza della propria azienda senza compromettere la funzione IBM MQ.

Non è necessario modificare le autorizzazioni e le proprietà dei file binari e delle librerie IBM MQ. La funzionalità di IBM MQ può risentirne a causa di questo tipo di modifica, in modo che i processi del gestore code potrebbero non riuscire ad accedere ad alcune delle risorse.

Tenere presente che le autorizzazioni e le proprietà non rappresentano alcuna minaccia per la sicurezza del sistema.

I dischi fissi Linux in cui è installato IBM MQ o in cui si trovano i dati IBM MQ non devono essere montati con l'opzione `nosuid`. Questa configurazione potrebbe inibire la funzione IBM MQ.

Per ulteriori informazioni, fare riferimento a [“Autorizzazioni del file system IBM MQ applicate a /var/mqm”](#) a pagina 65.

Concetti correlati

[File system](#)

Windows **Autorizzazioni del file system IBM MQ su Windows**

Le seguenti informazioni descrivono la sicurezza applicata ai file e alle directory su Windows. Per garantire il corretto funzionamento di IBM MQ, non è necessario modificare le autorizzazioni del file system come impostato da IBM MQ.

Directory di dati

Nota: Le autorizzazioni impostate sulla root di questa directory, vengono ereditate verso il basso in tutta la struttura di directory.

Le directory nella directory di dati (DATADIR) sono impostate con le autorizzazioni riportate di seguito, tranne le eccezioni descritte nel seguente testo.

Amministratori

Controllo completo

gruppo mqm

Controllo completo

SISTEMA

Controllo completo

Tutti

Lettura ed esecuzione

Le eccezioni sono:

DATADIR \errori

Tutti hanno il pieno controllo

DATADIR \traccia

Tutti hanno il pieno controllo

DATADIR \log

Amministratori

Controllo completo

gruppo mqm

Controllo completo

SISTEMA

Controllo completo

Tutti

Letto

DATADIR \log \ < qmgrname> \active

Amministratori

Controllo completo

gruppo mqm

Controllo completo

SISTEMA

Controllo completo

Nessun accesso concesso a tutti.

I file di log degli errori AMQERR01.LOG e così via, non ereditano le relative impostazioni di sicurezza dalla directory, ma sono invece impostate su Everyone: Full Control.

Release precedenti del prodotto

Nelle release del prodotto precedenti a IBM MQ 8.0, il programma predefinito e le directory di dati predefinite erano in co - ubicazione.

In qualsiasi installazione originariamente installata prima di IBM MQ 8.0. e che è stato installato nelle ubicazioni predefinite, e quindi aggiornato, le directory dei dati e del programma rimangono in co - ubicazione (in C:\Program Files\IBM\WebSphere MQ).

Nel caso di directory di dati e programmi co - ubicati, le precedenti informazioni si applicano solo alle directory che appartengono alla directory di dati e non a quelle che fanno parte della directory di programma.

Limitazioni di denominazione per code

Esistono restrizioni sulla lunghezza dei nomi delle code. Alcuni nomi coda sono riservati per le code definite dal gestore code.

Restrizioni sulle lunghezze dei nomi

Le code possono contenere un massimo di 48 caratteri.

Nomi coda riservati

I nomi che iniziano con "SYSTEM." sono riservati per le code definite dal gestore code. È possibile utilizzare i comandi **ALTER** o **DEFINE REPLACE** per modificare queste definizioni di coda in modo che si adattino alla propria installazione. I seguenti nomi sono definiti per IBM MQ:

Nome coda	Descrizione
SYSTEM.ADMIN.ACTIVITY.QUEUE	Coda per report di attività
SYSTEM.ADMIN.CHANNEL.EVENT	Coda per eventi canale
SYSTEM.ADMIN.COMMAND.EVENT	Coda per eventi comando
SYSTEM.ADMIN.COMMAND.QUEUE	Coda alla quale vengono inviati i messaggi di comando PCF
SYSTEM.ADMIN.CONFIG.EVENT	Coda per gli eventi di configurazione
SYSTEM.ADMIN.PERFM.EVENT	Coda per eventi di prestazioni
SYSTEM.ADMIN.PUBSUB.EVENT	Coda eventi correlati alla pubblicazione / sottoscrizione del sistema
SYSTEM.ADMIN.QMGR.EVENT	Coda per gli eventi del gestore code
SYSTEM.ADMIN.TRACE.ROUTE.QUEUE	Coda per messaggi di risposta di instradamento traccia
SYSTEM.AUTH.DATA.QUEUE	La coda che contiene gli elenchi di controllo accessi per il gestore code. (Non per z/OS)
SYSTEM.CHANNEL.INITQ	Coda di avvio per i canali

Tabella 41. Nomi e descrizioni delle code riservate (Continua)

Nome coda	Descrizione
SYSTEM.CHANNEL.SYNCQ	La coda che contiene i dati di sincronizzazione per i canali
SYSTEM.CHLAUTH.DATA.QUEUE	Coda dati di autenticazione canale IBM MQ
SYSTEM.CICS.INITIATION.QUEUE	Coda utilizzata per l'attivazione (non per z/OS)
SYSTEM.CLUSTER.COMMAND.QUEUE	Coda utilizzata per comunicare le modifiche del repository tra i gestori code
SYSTEM.CLUSTER.HISTORY.QUEUE	La coda viene utilizzata per memorizzare la cronologia delle informazioni sullo stato del cluster per scopi di servizio.
SYSTEM.CLUSTER.REPOSITORY.QUEUE	Coda utilizzata per conservare le informazioni sul repository
SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE	La coda utilizzata per creare singole code di trasmissione per ogni canale mittente del cluster.
SYSTEM.CLUSTER.TRANSMIT.QUEUE	La coda di trasmissione per tutte le destinazioni gestite dal supporto cluster
SYSTEM.COMMAND.INPUT	Coda a cui vengono inviati i messaggi di comando su z/OS
SYSTEM.COMMAND.REPLY.MODEL	Definizione di coda modello per le repliche di comando (per z/OS)
SYSTEM.DEAD.LETTER.QUEUE	Coda di messaggi non recapitabili (non per z/OS)
SYSTEM.DEFAULT.ALIAS.QUEUE	Definizione coda alias predefinita
SYSTEM.DEFAULT.INITIATION.QUEUE	Coda utilizzata per attivare un processo specifico (non per z/OS)
SYSTEM.DEFAULT.LOCAL.QUEUE	Definizione coda locale predefinita
SYSTEM.DEFAULT.MODEL.QUEUE	Definizione coda modello predefinita
SYSTEM.DEFAULT.REMOTE.QUEUE	Definizione coda remota predefinita
SYSTEM.DURABLE.SUBSCRIBER.QUEUE	Una coda locale utilizzata per conservare una copia persistente delle sottoscrizioni durevoli nel gestore code
SYSTEM.HIERARCHY.STATE	Coda utilizzata per congelare le informazioni sullo stato delle relazioni tra gestori code in una gerarchia di pubblicazione / sottoscrizione
SYSTEM.JMS.TEMPQ.MODEL	Modello per le code temporanee JMS
SYSTEM.INTERNAL.REPLY.QUEUE	Coda di risposta interna IBM MQ (non per z/OS)
SYSTEM.INTER.QMGR.CONTROL	Coda utilizzata in una gerarchia di pubblicazione / sottoscrizione per ricevere richieste da un gestore code remoto per creare una sottoscrizione proxy
SYSTEM.INTER.QMGR.PUBS	Coda utilizzata in una gerarchia di pubblicazione / sottoscrizione per ricevere pubblicazioni da un gestore code remoto
SYSTEM.INTER.QMGR.FANREQ	Coda utilizzata in una gerarchia di pubblicazione / sottoscrizione per elaborare richieste per creare una sottoscrizione proxy su un gestore code remoto

<i>Tabella 41. Nomi e descrizioni delle code riservate (Continua)</i>	
Nome coda	Descrizione
SYSTEM.MQEXPLORER.REPLY.MODEL	Definizione di coda modello per le risposte per IBM MQ Explorer
SYSTEM.MQSC.REPLY.QUEUE	Definizione di coda modello per le repliche del comando MQSC (non per z/OS)
SYSTEM.QSG.CHANNEL.SYNCQ	Coda locale condivisa utilizzata per memorizzare i messaggi che contengono le informazioni di sincronizzazione per i canali condivisi (solo z/OS)
SYSTEM.QSG.TRANSMIT.QUEUE	Coda locale condivisa utilizzata dall'agente di accodamento all'interno del gruppo durante la trasmissione di messaggi tra gestori code nello stesso gruppo di condivisione code (solo z/OS)
SYSTEM.RETAINED.PUB.QUEUE	Una coda locale utilizzata per conservare una copia di ciascuna pubblicazione conservata nel gestore code.
SYSTEM.SELECTION.EVALUATION.QUEUE	Coda di valutazione della selezione interna IBM MQ (non per z/OS)
SYSTEM.SELECTION.VALIDATION.QUEUE	IBM MQ coda di convalida di selezione interna (non per z/OS)

Limitazioni di denominazione per altri oggetti

Esistono delle limitazioni sulla lunghezza dei nomi oggetto. Alcuni nomi oggetto sono riservati per gli oggetti definiti dal gestore code.

Restrizioni sulla lunghezza del nome

I processi, gli elenchi dei nomi, i cluster, gli argomenti, i servizi e gli oggetti delle informazioni di autenticazione possono avere nomi lunghi fino a 48 caratteri.

I canali possono avere nomi lunghi fino a 20 caratteri.

Le classi di memoria possono avere nomi lunghi fino a 8 caratteri.


Le strutture CF possono avere nomi lunghi fino a 12 caratteri.

Nomi oggetto riservati

Nomi che cominciano con SYSTEM. sono riservati ad oggetti definiti dal gestore code. È possibile utilizzare i comandi **ALTER** o **DEFINE REPLACE** per modificare queste definizioni di oggetto in modo da adattare alla propria installazione. I seguenti nomi sono definiti per IBM MQ:

<i>Tabella 42. Descrizioni e nomi oggetto riservati</i>	
Nome oggetto	Descrizione
SYSTEM.ADMIN.SVRCONN	Canale di connessione server utilizzato per la gestione remota di un gestore code
SYSTEM.AUTO.RECEIVER	Canale ricevente predefinito per la definizione automatica (solo sistemi AIX, Linux, and Windows)
SYSTEM.AUTO.SVRCONN	Canale di connessione server predefinito per la definizione automatica (solo Multiplatforms)

Tabella 42. Descrizioni e nomi oggetto riservati (Continua)

Nome oggetto	Descrizione
SYSTEM.BASE.TOPIC	Argomento base per la risoluzione ASPARENT. Se un particolare oggetto argomento di gestione non ha alcun oggetto argomento di gestione principale, tutti gli attributi ASPARENT vengono ereditati da questo oggetto
SYSTEM.DEF.CLNTCONN	Definizione di canale di connessione client predefinita
SYSTEM.DEF.CLUSRCVR	Definizione di canale ricevente del cluster predefinita
SYSTEM.DEF.CLUSSDR	Definizione predefinita del canale mittente del cluster
SYSTEM.DEF.RECEIVER	Definizione canale ricevente predefinita
SYSTEM.DEF.REQUESTER	Definizione canale richiedente predefinita
SYSTEM.DEF.SENDER	Definizione canale mittente predefinita
SYSTEM.DEF.SERVER	Definizione di canale server predefinito
SYSTEM.DEF.SVRCONN	Definizione di canale di connessione server predefinita
SYSTEM.DEFAULT.AUTHINFO.CRLLDAP	Definizione dell'oggetto delle informazioni di autenticazione predefinito per la definizione degli oggetti delle informazioni di autenticazione di tipo CRLLDAP
SYSTEM.DEFAULT.AUTHINFO.OCSP	Definizione dell'oggetto delle informazioni di autenticazione predefinita per la definizione degli oggetti delle informazioni di autenticazione di tipo OCSP
SYSTEM.DEFAULT.LISTENER.LU62	Listener SNA predefinito (soloWindows)
SYSTEM.DEFAULT.LISTENER.NETBIOS	Listener NetBIOS predefinito (soloWindows)
SYSTEM.DEFAULT.LISTENER.SPX	Listener SPX predefinito (soloWindows)
SYSTEM.DEFAULT.LISTENER.TCP	Listener TCP/IP predefinito (solo Multiplatforms)
SYSTEM.DEFAULT.NAMELIST	Definizione elenco nomi predefinita
SYSTEM.DEFAULT.PROCESS	Definizione processo predefinita
SYSTEM.DEFAULT.SERVICE	Servizio predefinito (solo Multiplatforms)
SYSTEM.DEFAULT.TOPIC	Definizione argomento predefinita
SYSTEM.QPUBSUB.QUEUE.NAMELIST	Un elenco di code per l'interfaccia di pubblicazione / sottoscrizione accodata da monitorare
  SISTEMST	Definizione della classe di memorizzazione predefinita (soloz/OS)

Risoluzione nome coda

Nelle reti più grandi, l'uso dei gestori code presenta una serie di vantaggi rispetto ad altre forme di comunicazione. Questi vantaggi derivano dalla funzione di risoluzione dei nomi nella gestione delle code distribuite, che garantisce che la risoluzione dei nomi delle code venga eseguita dai gestori code sia all'invio che alla ricezione di un canale.

I principali vantaggi di questo approccio sono i seguenti:

- Le applicazioni non hanno bisogno di prendere decisioni di instradamento
- Le applicazioni non devono conoscere la struttura di rete

- I collegamenti di rete vengono creati dagli amministratori di sistema
- La struttura di rete è controllata dai pianificatori di rete
- È possibile utilizzare più canali tra i nodi per partizionare il traffico

La seguente figura mostra un esempio di risoluzione del nome della coda. La figura mostra due macchine in una rete, una che esegue un'applicazione put, l'altra che esegue un'applicazione get. Le applicazioni comunicano tra loro tramite il canale IBM MQ, controllato dagli MCA.

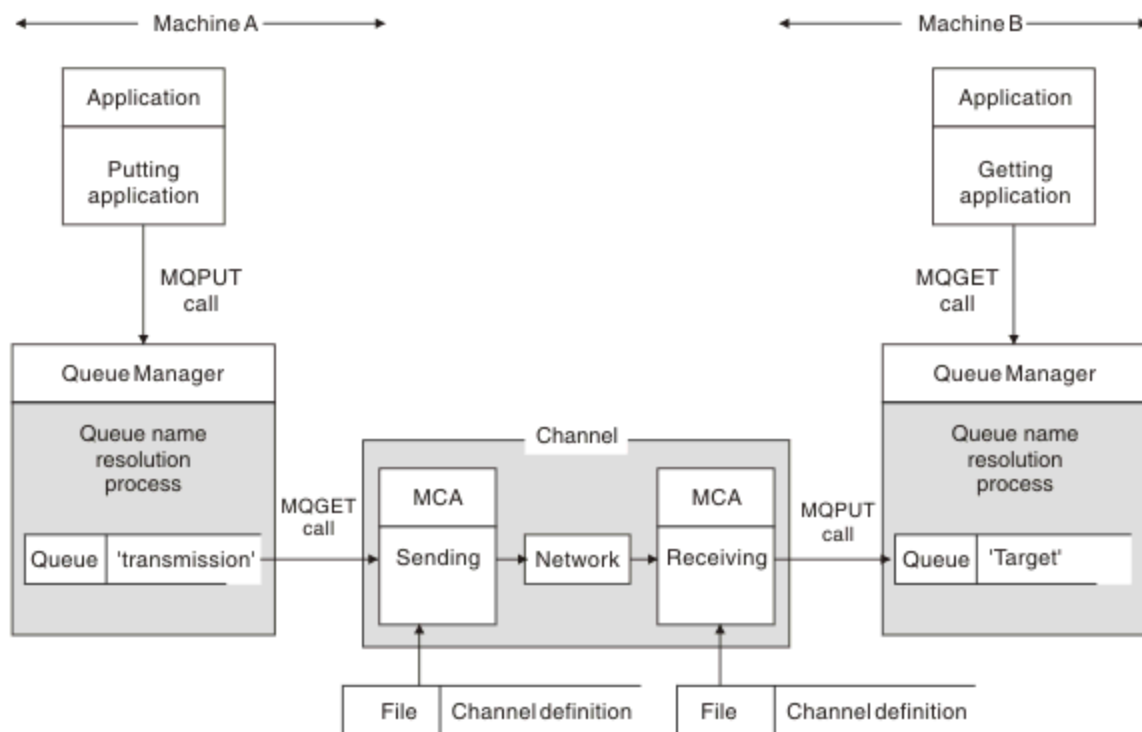


Figura 5. Risoluzione nomi

Facendo riferimento a [Figura 5 a pagina 76](#), il meccanismo di base per inserire i messaggi su una coda remota, per quanto riguarda l'applicazione, è lo stesso che per inserire i messaggi su una coda locale:

- L'applicazione che inserisce il messaggio emette chiamate MQOPEN e MQPUT per inserire messaggi nella coda di destinazione.
- L'applicazione che riceve i messaggi emette chiamate MQOPEN e MQGET per richiamare i messaggi dalla coda di destinazione.

Se entrambe le applicazioni sono connesse allo stesso gestore code, non è richiesta alcuna comunicazione tra gestori code e la coda di destinazione viene descritta come *locale* per entrambe le applicazioni.

Tuttavia, se le applicazioni sono connesse a gestori code differenti, due MCA e la loro connessione di rete associata sono coinvolti nel trasferimento, come mostrato nella figura. In questo caso, la coda di destinazione è considerata come una *coda remota* per l'applicazione di inserimento.

La sequenza di eventi è la seguente:

1. L'applicazione di inserimento emette chiamate MQOPEN e MQPUT per inserire messaggi nella coda di destinazione.
2. Durante la chiamata MQOPEN, la funzione *name resolution* rileva che la coda di destinazione non è locale e decide quale coda di trasmissione è appropriata. Successivamente, sulle chiamate MQPUT associate alla chiamata MQOPEN, tutti i messaggi vengono inseriti in questa coda di trasmissione.
3. L'MCA di invio richiama i messaggi dalla coda di trasmissione e li trasmette all'MCA di ricezione sul computer remoto.

4. L'MCA ricevente inserisce i messaggi nella coda di destinazione o nelle code.
5. L'applicazione di richiamo emette chiamate MQOPEN e MQGET per richiamare i messaggi dalla coda di destinazione.

Nota: Solo i passi 1 e 5 coinvolgono il codice dell'applicazione; i passi da 2 a 4 vengono eseguiti dai gestori code locali e dai programmi MCA. L'applicazione di inserimento ignora l'ubicazione della coda di destinazione, che potrebbe trovarsi nello stesso processore o in un altro processore su un altro continente.

La combinazione dell'MCA di invio, della connessione di rete e dell'MCA di ricezione viene chiamata *canale di messaggied* è intrinsecamente una periferica unidirezionale. Normalmente, è necessario spostare i messaggi in entrambe le direzioni, e due canali sono impostati per questo movimento, uno in ogni direzione.

Attività correlate

Inserimento di messaggi nelle code remote

Qual è la risoluzione del nome della coda?

La risoluzione del nome della coda è fondamentale per la gestione della coda distribuita. Elimina la necessità che le applicazioni siano interessate all'ubicazione fisica delle code e isola le applicazioni dai dettagli delle reti.

Un amministratore di sistema può spostare le code da un gestore code a un altro e modificare l'instradamento tra i gestori code senza che le applicazioni debbano essere informate.

Per disaccoppiare dall'applicazione il percorso esatto su cui viaggiano i dati, esiste un livello di riferimento indiretto tra il nome utilizzato dall'applicazione quando fa riferimento alla coda di destinazione e il nome del canale su cui si verifica il flusso. Questa via indiretta viene ottenuta utilizzando il meccanismo di risoluzione del nome coda.

In sostanza, quando un'applicazione fa riferimento a un nome coda, il nome viene associato dal meccanismo di risoluzione a una coda di trasmissione o a una coda locale che non è una coda di trasmissione. Per l'associazione ad una coda di trasmissione, è necessaria una seconda risoluzione del nome nella destinazione e il messaggio ricevuto viene inserito nella coda di destinazione come previsto dal progettista dell'applicazione. L'applicazione rimane ignora della coda di trasmissione e dei canali utilizzati per spostare il messaggio.

Nota: La definizione della coda e del canale è una responsabilità di gestione del sistema e può essere modificata da un operatore o da un programma di utilità di gestione del sistema, senza la necessità di modificare le applicazioni.

Un requisito importante per la gestione del sistema dei flussi di messaggi è la necessità di fornire percorsi alternativi tra gestori code. Ad esempio, i requisiti aziendali potrebbero richiedere che le diverse *classi di servizio* vengano inviate attraverso canali differenti alla stessa destinazione. Questa decisione è una decisione di gestione del sistema e il meccanismo di risoluzione del nome della coda fornisce un modo flessibile per ottenerlo. Il manuale Application Programming Guide lo descrive in dettaglio, ma l'idea di base è di utilizzare la risoluzione del nome della coda sul gestore code di invio per associare il nome della coda fornito dall'applicazione alla coda di trasmissione appropriata per il tipo di traffico coinvolto. Analogamente, all'estremità di ricezione, la risoluzione del nome della coda associa il nome nel descrittore del messaggio ad una coda locale (non di trasmissione) o di nuovo ad una coda di trasmissione appropriata.

Non solo è possibile partizionare il percorso di inoltro da un gestore code a un altro in diversi tipi di traffico, ma il messaggio di ritorno inviato alla definizione della coda reply - to nel messaggio in uscita può anche utilizzare lo stesso partizionamento del traffico. La risoluzione del nome coda soddisfa questo requisito e il progettista dell'applicazione non deve essere coinvolto in queste decisioni di partizionamento del traffico.

Il punto in cui l'associazione viene eseguita sia sui gestori code di invio che su quelli di ricezione è un aspetto importante della modalità di funzionamento della risoluzione dei nomi. Questa associazione consente al nome coda fornito dall'applicazione di inserimento di essere associato a una coda locale o a

una coda di trasmissione sul gestore code di invio e di nuovo riassociato a una coda locale o a una coda di trasmissione sul gestore code di ricezione.

I messaggi di risposta dalle applicazioni di ricezione o dagli MCA hanno la risoluzione del nome eseguita nello stesso modo, consentendo l'instradamento di ritorno su percorsi specifici con definizioni di coda in tutti i gestori code sull'instradamento.

Come vengono risolti gli attributi degli oggetti di destinazione per gli alias, le code remote e le code cluster?

Quando la risoluzione dei nomi viene eseguita per conto di una chiamata dell'API dell'applicazione, gli attributi che influenzano l'utilizzo dell'oggetto vengono risolti da una combinazione dell'oggetto denominato in origine, del "percorso" (consultare ["Risoluzione nome coda"](#) a pagina 75) e dell'oggetto di destinazione risolto. In un cluster di gestori code, l'"oggetto denominato" in questione è la definizione dell'oggetto cluster (coda o argomento). Questo è un sottoinsieme di attributi oggetto condivisi tra gestori code e visibili attraverso. ad esempio, **DISPLAY QCLUSTER**.

Quando un attributo può essere definito sull'oggetto denominato aperto dall'applicazione, questo ha la precedenza. Ad esempio, tutti gli attributi DEF* *** (persistenza predefinita, priorità e risposta di inserimento asincrono) possono essere configurati su definizioni di code remote e alias. Queste diventano effettive quando l'alias o la coda remota viene aperta da un'applicazione, piuttosto che da qualsiasi coda di destinazione o coda di trasmissione risolta.

Gli attributi progettati per limitare o limitare l'interazione dell'applicazione con un oggetto di destinazione non possono in genere essere definiti sull'oggetto denominato (definizione di coda remota o alias). Ad esempio, **MAXMSGL** e **MAXDEPTH** non possono essere impostati su una definizione di coda remota o su un alias e non vengono passati tra membri di un cluster di gestori code. Questi attributi vengono quindi presi dalla coda risolta (ad esempio, la coda locale, la coda di trasmissione appropriata o SYSTEM.CLUSTER.TRANSMIT.QUEUE). All'arrivo in un gestore code remoto, un secondo vincolo potrebbe essere applicato al recapito alla coda di destinazione, il che potrebbe causare l'inserimento di un messaggio in una coda di messaggi non recapitabili o l'arresto forzato del canale.

Si noti che un caso speciale di risoluzione attributo è l'abilitazione **PUT** e **GET**. Per entrambi questi attributi, qualsiasi istanza di **DISABLED** nel percorso della coda risulta in un attributo globale risolto di **DISABLED**.

Oggetti di sistema e predefiniti

Elenca gli oggetti predefiniti e di sistema creati dal comando **crtmqm**.

Quando si crea un gestore code utilizzando il comando di controllo **crtmqm**, gli oggetti di sistema e gli oggetti predefiniti vengono creati automaticamente.

- Gli oggetti di sistema sono quegli oggetti IBM MQ necessari per gestire un gestore code o un canale.
- Gli oggetti predefiniti definiscono tutti gli attributi di un oggetto. Quando si crea un oggetto, ad esempio una coda locale, tutti gli attributi non specificati esplicitamente vengono ereditati dall'oggetto predefinito.

Le tabelle riportate di seguito elencano gli oggetti predefiniti e di sistema creati da **crtmqm**.

Nota: Esistono altri due oggetti predefiniti non inclusi nelle tabelle: l'oggetto gestore code e il catalogo oggetti. Questi sono oggetti nel senso che sono registrati e recuperabili.

- [Sistema e oggetti predefiniti: code](#)
- [Sistema e oggetti predefiniti: argomenti](#)
- [Sistema e oggetti predefiniti: canali server](#)
- [Sistema e oggetti predefiniti: canali client](#)
- [Oggetti di sistema e predefiniti: informazioni di autenticazione](#)
- [Sistema e oggetti predefiniti: informazioni sulle comunicazioni](#)

- [Sistema e oggetti predefiniti: listener](#)
- [Sistema e oggetti predefiniti: elenchi nomi](#)
- [Sistema e oggetti predefiniti: processi](#)
- [Sistema e oggetti predefiniti: servizi](#)

<i>Tabella 43. Oggetti di sistema e predefiniti: code</i>	
Nome oggetto	Descrizione
SYSTEM.ADMIN.ACCOUNTING.QUEUE	Coda per i dati del messaggio di account generati quando un'applicazione si disconnette dal gestore code.
SYSTEM.ADMIN.ACTIVITY.QUEUE	Coda che contiene i messaggi di report di attività restituiti.
SYSTEM.ADMIN.CHANNEL.EVENT	Coda eventi per i canali.
SYSTEM.ADMIN.COMMAND.EVENT	Coda eventi per eventi comando.
SYSTEM.ADMIN.COMMAND.QUEUE	Coda comandi di gestione. Utilizzato per comandi MQSC e PCF remoti.
SYSTEM.ADMIN.CONFIG.EVENT	Coda eventi per eventi di configurazione.
SYSTEM.ADMIN.LOGGER.EVENT	Coda eventi per i messaggi di evento del programma di registrazione (ricevitore di giornale).
SYSTEM.ADMIN.PERFM.EVENT	Coda eventi per eventi di prestazioni.
SYSTEM.ADMIN.PUBSUB.EVENT	Coda eventi correlati alla pubblicazione / sottoscrizione del sistema
SYSTEM.ADMIN.QMGR.EVENT	Coda eventi per gli eventi del gestore code.
SYSTEM.ADMIN.STATISTICS.QUEUE	La coda che contiene i dati di monitoraggio delle statistiche MQI, coda e canale.
SYSTEM.ADMIN.TRACE.ACTIVITY.QUEUE	La coda che visualizza l'attività di traccia.
SYSTEM.ADMIN.TRACE.ROUTE.QUEUE	La coda che contiene i messaggi di risposta di trace - route restituiti.
SYSTEM.AMQP.COMMAND.QUEUE	Coda dei comandi di amministrazione IBM MQ per AMQP
SYSTEM.AUTH.DATA.QUEUE	La coda che contiene gli elenchi di controllo accessi per il gestore code. Utilizzato da OAM (object authority manager).
SYSTEM.BROKER.ADMIN.STREAM	Flusso di gestione per interfaccia Pub / Sot accodata
SYSTEM.BROKER.CONTROL.QUEUE	Coda di controllo interfaccia di pubblicazione / sottoscrizione.
SYSTEM.BROKER.DEFAULT.STREAM	Flusso predefinito per interfaccia Pub / Sot accodata
SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS	Broker per la coda di comunicazioni broker.
SYSTEM.CHANNEL.INITQ	Coda di iniziazione canale.
SYSTEM.CHANNEL.SYNCQ	La coda che contiene i dati di sincronizzazione per canali.
SYSTEM.CHLAUTH.DATA.QUEUE	Coda dati di autenticazione canale IBM MQ

Tabella 43. Oggetti di sistema e predefiniti: code (Continua)

Nome oggetto	Descrizione
SYSTEM.CICS.INITIATION.QUEUE	La coda di avvio CICS predefinita.
SYSTEM.CLUSTER.COMMAND.QUEUE	La coda utilizzata per trasportare i messaggi al gestore code del repository.
SYSTEM.CLUSTER.HISTORY.QUEUE	La coda utilizzata per memorizzare la cronologia delle informazioni sullo stato del cluster per scopi di servizio.
SYSTEM.CLUSTER.REPOSITORY.QUEUE	La coda utilizzata per memorizzare tutte le informazioni del repository.
SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE	La coda utilizzata per creare singole code di trasmissione per ogni canale mittente del cluster.
SYSTEM.CLUSTER.TRANSMIT.QUEUE	La coda di trasmissione per tutti i messaggi a tutti i cluster.
SYSTEM.DEAD.LETTER.QUEUE	Coda di lettere non recapitate (messaggio non consegnato).
SYSTEM.DEFAULT.ALIAS.QUEUE	Coda alias predefinita.
SYSTEM.DEFAULT.INITIATION.QUEUE	Coda di iniziazione predefinita.
SYSTEM.DEFAULT.LOCAL.QUEUE	Coda locale predefinita.
SYSTEM.DEFAULT.MODEL.QUEUE	Coda modello predefinita.
SYSTEM.DEFAULT.REMOTE.QUEUE	Coda remota predefinita.
SYSTEM.DOTNET.XARECOVERY.QUEUE	Coda di ripristino IBM MQ .NET XA
SYSTEM.DURABLE.MODEL.QUEUE	La coda utilizzata come modello per le sottoscrizioni durevoli gestite.
SYSTEM.DURABLE.SUBSCRIBER.QUEUE	La coda utilizzata per conservare una copia persistente delle sottoscrizioni durevoli nel gestore code.
SYSTEM.HIERARCHY.STATE	IBM MQ stato della relazione della gerarchia di pubblicazione / sottoscrizione distribuita.
SYSTEM.INTER.QMGR.CONTROL	Coda di controllo di pubblicazione / sottoscrizione distribuita IBM MQ .
SYSTEM.INTER.QMGR.FANREQ	Coda di input del processo fan - out della sottoscrizione proxy interna di pubblicazione / sottoscrizione distribuita IBM MQ .
SYSTEM.INTER.QMGR.PUBS	Pubblicazioni di pubblicazione / sottoscrizione distribuite IBM MQ .
SYSTEM.INTERNAL.REPLY.QUEUE	
SYSTEM.INTERNAL.REQUEST.QUEUE	
SYSTEM.JMS.TEMPQ.MODEL	Modello per le code temporanee JMS
SYSTEM.MQEXPLORER.REPLY.MODEL	La coda di risposta IBM MQ Explorer . Questa è una coda modello che crea una coda dinamica temporanea per le risposte a IBM MQ Explorer.

Tabella 43. Oggetti di sistema e predefiniti: code (Continua)

Nome oggetto	Descrizione
SYSTEM.MQSC.REPLY.QUEUE	Coda di risposta comando MQSC. Questa è una coda modello che crea una coda dinamica temporanea per le risposte ai comandi MQSC remoti.
SYSTEM.NDURABLE.MODEL.QUEUE	Una coda utilizzata come modello per sottoscrizioni non durevoli gestite.
SYSTEM.PENDING.DATA.QUEUE	Supporta i messaggi differiti in JMS.
SYSTEM.PROTECTION.ERROR.QUEUE	IBM MQ Coda di errori di protezione messaggi.
SYSTEM.PROTECTION.POLICY.QUEUE	IBM MQ Coda della politica di protezione dei messaggi.
SYSTEM.REST.REPLY.QUEUE	
SYSTEM.RETAINED.PUB.QUEUE	Una coda utilizzata per conservare una copia di ciascuna pubblicazione conservata nel gestore code.
SYSTEM.SELECTION.EVALUATION.QUEUE	
SYSTEM.SELECTION.VALIDATION.QUEUE	

Tabella 44. Oggetti di sistema e predefiniti: argomenti

Nome oggetto	Descrizione
SYSTEM.ADMIN.TOPIC	Argomento di amministrazione.
SYSTEM.BASE.TOPIC	Argomento di base per la risoluzione ASPARENT . Se un particolare argomento non dispone di oggetti argomento di gestione principale o se tali oggetti principali dispongono anche di ASPARENT, qualsiasi attributo ASPARENT rimanente viene ereditato da questo oggetto.
SYSTEM.BROKER.ADMIN.STREAM	Flusso di gestione utilizzato dall'interfaccia di pubblicazione / sottoscrizione in coda.
SYSTEM.BROKER.DEFAULT.STREAM	Il flusso predefinito utilizzato dall'interfaccia di pubblicazione / sottoscrizione in coda.
SYSTEM.BROKER.DEFAULT.SUBPOINT	Il punto secondario predefinito utilizzato dall'interfaccia di pubblicazione / sottoscrizione in coda.
SYSTEM.DEFAULT.TOPIC	Definizione argomento predefinita.

Tabella 45. Oggetti di sistema e predefiniti: canali server

Nome oggetto	Descrizione
SYSTEM.AUTO.RECEIVER	Canale ricevente dinamico.
SYSTEM.AUTO.SVRCONN	Canale di connessione server dinamico.
SYSTEM.DEF.AMQP	Canale AMQP predefinito. Notare che l'oggetto è definito, ma il servizio AMQP non è supportato.

Tabella 45. Oggetti di sistema e predefiniti: canali server (Continua)

Nome oggetto	Descrizione
SYSTEM.DEF.CLUSRCVR	Canale ricevente predefinito per il cluster, utilizzato per fornire i valori predefiniti per gli attributi non specificati quando un canale CLUSRCVR viene creato su un gestore code nel cluster.
SYSTEM.DEF.CLUSSDR	Canale mittente predefinito per il cluster, utilizzato per fornire i valori predefiniti per gli attributi non specificati quando un canale CLUSSDR viene creato su un gestore code nel cluster.
SYSTEM.DEF.RECEIVER	Canale ricevente predefinito.
SYSTEM.DEF.REQUESTER	Canale richiedente predefinito.
SYSTEM.DEF.SENDER	Canale mittente predefinito.
SYSTEM.DEF.SERVER	Canale server predefinito.
SYSTEM.DEF.SVRCONN	Canale di connessione server predefinito.
SYSTEM.DEFAULT.AUTHINFO.IDPWLDAP	
SYSTEM.DEFAULT.AUTHINFO.IDPWOS	

Tabella 46. Oggetti predefiniti e di sistema: canali client

Nome oggetto	Descrizione
SYSTEM.DEF.CLNTCONN	Canale di connessione client predefinito.

Tabella 47. Oggetti di sistema e predefiniti: informazioni di autenticazione

Nome oggetto	Descrizione
SYSTEM.DEFAULT.AUTHINFO.CRLLDAP	L'oggetto delle informazioni di autenticazione predefinito per la definizione degli oggetti delle informazioni di autenticazione di tipo CRLLDAP.
SYSTEM.DEFAULT.AUTHINFO.OCSP	Oggetto delle informazioni di autenticazione predefinito per la definizione degli oggetti delle informazioni di autenticazione di tipo OCSP.

Tabella 48. Oggetti di sistema e predefiniti: informazioni sulle comunicazioni

Nome oggetto	Descrizione
SYSTEM.DEFAULT.COMMINFO.MULTICAST	Oggetto informazioni di comunicazione predefinito per multicast.

Tabella 49. Oggetti predefiniti e di sistema: listener


Nome oggetto	Descrizione
SYSTEM.DEFAULT.LISTENER.TCP	Listener predefinito per il trasporto TCP.
 SYSTEM.DEFAULT.LISTENER.LU62	Listener LU62 predefinito.

Tabella 49. Oggetti predefiniti e di sistema: listener (Continua)



Nome oggetto	Descrizione
 SYSTEM.DEFAULT.LISTENER.NETBIOS	Listener NETBIOS predefinito.
 SYSTEM.DEFAULT.LISTENER.SPX	Listener SPX predefinito.

Tabella 50. Oggetti di sistema e predefiniti: elenchi nomi

Nome oggetto	Descrizione
SYSTEM.DEFAULT.NAMELIST	Definizione elenco nomi predefinita.
SYSTEM.QPUBSUB.QUEUE.NAMELIST	Un elenco di nomi di coda monitorati dall'interfaccia di pubblicazione / sottoscrizione in coda.
SYSTEM.QPUBSUB.SUBPOINT.NAMELIST	Un elenco di oggetti argomento utilizzati dall'interfaccia di pubblicazione / sottoscrizione in coda per mettere in corrispondenza gli oggetti argomento con i punti di sottoscrizione.

Tabella 51. Oggetti di sistema e predefiniti: processi

Nome oggetto	Descrizione
SYSTEM.DEFAULT.PROCESS	Definizione processo predefinita.

Tabella 52. Oggetti di sistema e predefiniti: servizi

Nome oggetto	Descrizione
SYSTEM.AMQP.SERVICE	Servizio API MQ Light . Notare che l'oggetto è definito, ma il servizio non è supportato.
SYSTEM.DEFAULT.SERVICE	Servizio predefinito.

SYSTEM.BASE.TOPIC

Argomento di base per la risoluzione ASPARENT . Se un particolare argomento non dispone di oggetti argomento di gestione principale o se tali oggetti principali dispongono anche di ASPARENT, qualsiasi attributo ASPARENT rimanente viene ereditato da questo oggetto.

Tabella 53. Valori predefiniti di SYSTEM.BASE.TOPIC




Parametro	Valore
TOPICSTR	"
 CAPEXPY	NOLIMIT
CLROUTE	DIRECT
cluster	Il valore predefinito è una stringa vuota.
COMMINFO	SYSTEM.DEFAULT.COMMINFO.MULTICAST
DEFPRESP	SYNC
DEFPRTY	0

Tabella 53. Valori predefiniti di SYSTEM.BASE.TOPIC (Continua)

Parametro	Valore
DEFPSIST	NO
DESCR	'Argomento di base per la risoluzione degli attributi'
DURSUB	YES
MCAST	DISABLED
MDURMDL	SYSTEM.DURABLE.MODEL.QUEUE
MNDURMDL	SYSTEM.NDURABLE.MODEL.QUEUE
NPMSGDLV	ALLAVAIL
PMSGDLV	ALLDUR
PROXYSUB	FIRSTUSE
PUB	ENABLED
PUBSCOPE	ALL
  QSGDISP (solo piattaforma z/OS)	QMGR
SUB	ENABLED
SUBSCOPE	ALL
USEDLQ	YES
WILDCARD	PASSTHRU

Se questo oggetto non esiste, i suoi valori predefiniti vengono ancora utilizzati da IBM MQ per attributi ASPARENT che non vengono risolti dagli argomenti principali più in alto nella struttura ad albero degli argomenti.

L'impostazione degli attributi PUB o SUB di SYSTEM.BASE.TOPIC su DISABLED impedisce alle applicazioni di pubblicare o sottoscrivere argomenti nella struttura ad albero degli argomenti, con due eccezioni:

1. Tutti gli oggetti argomento nella struttura ad albero degli argomenti che hanno PUB o SUB esplicitamente impostato su ENABLE. Le applicazioni possono pubblicare o sottoscrivere tali argomenti e i relativi child.
2. La pubblicazione e la sottoscrizione a SYSTEM.BROKER.ADMIN.STREAM non sono disabilitate dall'impostazione degli attributi PUB o SUB di da SYSTEM.BASE.TOPIC a DISABLED.

Consultare anche [Gestione speciale per il parametro PUB](#).






Informazioni sulla stanza dei file di configurazione

Le seguenti informazioni consentono di configurare le informazioni nelle stanze ed elencare il contenuto dei file `mqs.ini`, `qm.inie` `mqclient.ini`.

Configurazione delle stanze

Utilizzare i collegamenti per configurare il sistema, o i sistemi, nell'azienda:

- Le stanze e gli attributi del file `mqs.ini` consentono di configurare:
 - Stanza *AllQueueManagers*
 - Stanza *DefaultQueueManager*

- Stanza *ExitProperties*
- Stanza *LogDefaults*
- Stanza *Security* nel file *qm.ini*
- Le stanze e gli attributi del file *qm.ini* consentono di configurare:
 -  Stanza *AccessMode* (soloWindows)
 - Stanza *Service* - per i servizi installabili
 - Stanza *Log*
 -   Stanza *RestrictedMode* (solo sistemiAIX and Linux)
 - Stanza *XAResourceManager*
 - Stanze *TCP*, *LU62* e *NETBIOS*
 - Stanza *ExitPath*
 - Stanza *QMErrorLog*
 - Stanza *SSL*
 - Stanza *ExitPropertiesLocal*
- Configurazione di servizi e componenti consente di configurare:
 - Stanza *Service*
 - Stanza *ServiceComponent*
 e contiene link al modo in cui vengono utilizzati per diversi servizi su piattaforme AIX, Linux, and Windows .
- Configurazione delle uscite API consente di configurare:
 - Stanza *AllActivityTrace*
 - Stanza *ApplicationTrace*
- Configurazione del comportamento della traccia attività consente di configurare:
 - Stanza *ApiExitCommon*
 - stanza *ApiExitTemplate*
 - Stanza *ApiExitlocale*
- IBM MQ MQI client file di configurazione, *mqclient.ini* consente di configurare:
 - stanza *CHANNELS*
 - Stanza *ClientExitPath*
 -  *LU62*, stanza *NETBIOS* e *SPX* (soloWindows)
 - Stanza *MessageBuffer*
 - Stanza *SSL*
 - Stanza *TCP*
 -  Stanza *Trace* (utilizzata solo per IBM MQ .NET e XMS .NET)
- “Stanze del file di configurazione per l'accodamento distribuito” a pagina 87 consente di configurare:
 - stanza *CHANNELS*
 - Stanza *TCP*
 - Stanza *LU62*
 - *NETBIOS*
 - Stanza *ExitPath*
- Impostazione degli attributi dei messaggi di pubblicazione / sottoscrizione accodati consente di configurare:

- Attributo *PersistentPublishRiprova*
- attributo *NonPersistentPublishRetry*
- Attributo *PublishBatchDimensione*
- Attributo *PublishRetryPublishRetry*

nella stanza *Broker* .



Attenzione: È necessario creare una stanza *Broker* , se ne è necessaria una.

- L'uso della configurazione automatica consente di configurare:
 - Stanza *AutoConfig*
 - Stanza *AutoCluster*
 - Stanza delle variabili

File di configurazione

Consultare:

- [Filemq.s.ini](#)
- File [qm.ini](#)
- [mqclient.ini](#) file

per un elenco delle possibili stanze in ogni file di configurazione.

Linux

AIX

File mq.s.ini

L'esempio di un IBM MQ file di configurazione per sistemi AIX and Linux mostra un file `mq.s.ini` di esempio.

Un file `mq.s.ini` può contenere le seguenti stanze:

- [AllQueueAllQueue](#)
- [GestoreDefaultQueue](#)
- [ExitProperties](#)
- [LogDefaults](#)

Inoltre, è presente una stanza [QueueManager](#) per ciascun gestore code.

File qm.ini

Esempio di file di configurazione del gestore code per i IBM MQ for AIX or Linux sistemi mostra un esempio di file `qm.ini` .

Un file `qm.ini` può contenere le seguenti stanze:

- [ExitPath](#)
- [Log](#)
- [QMErrorLog](#)
- [QueueManager](#)
- [Sicurezza](#)
- [ServiceComponent](#)

Multi

Per configurare [InstallableServices](#) utilizzare le stanze *Service* e *ServiceComponent* .

- Connessione per [DefaultBindTipo](#)





Attenzione: È necessario creare una stanza *Connection* , se ne è necessaria una.

- [SSL e TLS](#)

- [TCP, LU62e NETBIOS](#)
- [XAResourceManager](#)

Inoltre, è possibile modificare:

-  [AccessMode](#) (soloWindows)
-  [RestrictedMode](#) (solo sistemiAIX and Linux)

utilizzando il comando [crtmqm](#) .

File mqclient.ini

Un file mqclient.ini può contenere le seguenti stanze:

- [CANALI](#)
- [ClientExitClientExit](#)
- [LU62, NETBIOS e SPX](#)
- [MessageBuffer](#)
- [SSL](#)
- [TCP](#)

Inoltre, potrebbe essere necessaria una stanza [PreConnect](#) per configurare un'uscita di preconnessione.

Stanze del file di configurazione per l'accodamento distribuito

Una descrizione delle stanze del file di configurazione del gestore code, qm.ini, relativo alla coda distribuita.

Questo argomento mostra le stanze nel file di configurazione del gestore code relative all'accodamento distribuito. Viene applicato al file di configurazione del gestore code per IBM MQ for Multiplatforms. Il file è denominato qm.ini su tutte le piattaforme.

Le stanze relative all'accodamento distribuito sono:

- Canali
- TCP
- LU62
- NETBIOS
- PERCORSO

[Figura 6 a pagina 88](#) mostra i valori che è possibile impostare utilizzando queste stanze. Quando si definisce una di queste stanze, non è necessario avviare ogni elemento su una nuova riga. È possibile utilizzare un punto e virgola (;) o un carattere cancelletto (#) per indicare un commento.

```

CHANNELS:
MAXCHANNELS=n           ; Maximum number of channels allowed, the
                        ; default value is 100.
MAXACTIVECHANNELS=n    ; Maximum number of channels allowed to be active at
                        ; any time, the default is the value of MaxChannels.
MAXINITIATORS=n        ; Maximum number of initiators allowed, the default
                        ; and maximum value is 3.
MQIBINDTYPE=type       ; Whether the binding for applications is to be
                        ; "fastpath" or "standard".
                        ; The default is "standard".
PIPELINELENGTH=n       ; The maximum number of concurrent threads a channel will use.
                        ; The default is 1. Any value greater than 1 is treated as 2.
ADOPTNEWMCA=chltype    ; Stops previous process if channel fails to start.
                        ; The default is "NO".
ADOPTNEWMCATIMEOUT=n   ; Specifies the amount of time that the new
                        ; process should wait for the old process to end.
                        ; The default is 60.
ADOPTNEWMCACHECHECK=   ; Specifies the type checking required.
  typecheck             ; The default is "NAME", "ADDRESS", and "QM".
CHLAUTHEARLYADOPT=Y/N ; The order in which connection authentication and channel
authentication rules are ; processed. If not present in the qm.ini file the default is "N".
From MQ9.0.4 all
PASSWORDPROTECTION=    ; queue managers are created with a default of "Y"
than using TLS.        ; From MQ8.0, set protected passwords in the MQCSP structure, rather
  options              ; The options are "compatible", "always", "optional" and "warn"
                        ; The default is "compatible".
IGNORESEQNUMBERMISMATCH ; How the queue manager handles a sequence number mismatch during
channel startup.      ;
  =Y/N                ; The options are "Y" and "N" with the default being "N".
CHLAUTHIGNOREUSERCASE  ; Enables a queue manager to make username matching within CHLAUTH
rules case-insensitive. ;
  =Y/N                ; The options are "Y" and "N" with the default being "N".
CHLAUTHISSUEWARN=Y     ; If you want message AMQ9787 to be generated when you set theWARN=YES
attribute              ;
                        ; on the SET CHLAUTH command.
TCP:                   ; TCP entries
  PORT=n               ; Port number, the default is 1414
  KEEPALIVE=Yes        ; Switch TCP/IP KeepAlive on
LU62:
  LIBRARY2=DLLName2   ; Used if code is in two libraries
EXITPATH:1 Location of user exits
EXITPATHS=            ; String of directory paths.

```

Figura 6. Stanze qm.ini per l'accodamento distribuito

Note:

1. EXITPATH si applica solo alle piattaforme seguenti:

-  AIX
-  Windows

Attività correlate

[Configurazione](#)

 [Configurazione di z/OS](#)

[Modifica delle informazioni di configurazione IBM MQ su Multiplatforms](#)

Attributi canale

Questa sezione descrive gli attributi del canale contenuti nelle definizioni di canale.

È possibile scegliere gli attributi di un canale in modo che siano ottimali per una particolare serie di circostanze per ciascun canale. Tuttavia, quando il canale è in esecuzione, i valori effettivi potrebbero essere stati modificati durante le negoziazioni di avvio. Consultare [Preparazione dei canali](#).

Molti attributi hanno valori predefiniti ed è possibile utilizzarli per la maggior parte dei canali. Tuttavia, nelle circostanze in cui i valori predefiniti non sono ottimali, consultare questa sezione per una guida nella selezione dei valori corretti.

Nota: In IBM MQ for IBM i, la maggior parte degli attributi può essere specificata come *SYSDFTCHL, il che significa che il valore viene preso dal canale predefinito di sistema nel sistema.

I tipi di canale per gli attributi del canale IBM MQ sono elencati nella seguente tabella, nell'ordine dei parametri del comando MQSC.

Nota: Per i canali del cluster (colonne CLUSSDR e CLUSRCVR nella tabella), se è possibile impostare un attributo su entrambi i canali, impostarlo e accertarsi che le impostazioni siano identiche. Se vi è una discrepanza tra le impostazioni, è possibile che vengano utilizzate quelle specificate sul canale CLUSRCVR. Questo è spiegato in [Canali cluster](#).

Tabella 54. Attributi del canale per i tipi di canali

Campo attributo	Parametro del comando MQSC	SDR	SVR	RCV R	RQST R	CONN CLNT	SVR - CONN	CLUS - SDR	CLUS - RCVR	AMQP
Affinità connessione	AFFINITÀ					Sì				
Data modifica	ALTDATE	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì
Modifica ora	ALTTIME	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì
Multi alive AMQP <small>keep</small>	AMQPKA									Sì
Intervallo heartbeat batch	BATCHHB	Sì	Sì					Sì	Sì	
Intervallo batch	BATCHINT	Sì	Sì					Sì	Sì	
Limite batch	BATCHLIM	Sì	Sì					Sì	Sì	
dimensione batch	BATCHSZ	Sì	Sì	Sì	Sì			Sì	Sì	
Etichetta certificato	CERTLABL	Sì	Sì	Sì	Sì	Sì	Sì	Sì ^{"1"} a pagina 93	Sì	Sì
Nome canale	CHANNEL	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì
Tipo canale	CHLTYPE	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì
Importanza canale client	CLNTWGH T					Sì				
Elenco nomi cluster	CLUSNL							Sì	Sì	
Cluster	cluster							Sì	Sì	
Priorità carico di lavoro cluster	CLWLPRTY							Sì	Sì	
Classificazione carico di lavoro cluster	CLWLRRANK							Sì	Sì	
Peso carico di lavoro cluster	CLWLWGH T							Sì	Sì	
Compressione intestazione	COMPHDR	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	

Tabella 54. Attributi del canale per i tipi di canali (Continua)

Campo attributo	Parametro del comando MQSC	SDR	SVR	RCV R	RQST R	CONN CLNT	SVR - CONN	CLUS - SDR	CLUS - RCVR	AM QP
<u>Compressione dei dati</u>	COMPMSG	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	
<u>Nome connessione</u>	CONNNAME	Sì	Sì		Sì	Sì		Sì	Sì	
<u>Conversione messaggio</u>	CONVERT	Sì	Sì					Sì	Sì	
<u>Riconnessione predefinita</u>	DEFRECON					Sì				
<u>Descrizione</u>	DESCR	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì
<u>Intervallo di disconnessione</u>	DISCINT	Sì	Sì				z/OS Sì Multi No	Sì	Sì	
<u>intervallo di heartbeat</u>	HBINT	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	
<u>Intervallo di keepalive</u>	KAINT	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	
<u>Indirizzo locale</u>	LOCLADDR	Sì	Sì		Sì	Sì		Sì	Sì	Sì
<u>Numero di tentativi lunghi</u>	LONGRTY	Sì	Sì					Sì	Sì	
<u>Intervallo nuovo tentativo lungo</u>	LONGTMR	Sì	Sì					Sì	Sì	
<u>Istanze massime</u>	MAXINST						Sì			Sì
<u>Massimo istanze per client</u>	MAXINSTC						Sì			
<u>Lunghezza massima dei messaggi</u>	MAXMSGLEN	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì
<u>Nome MCA</u>	MCANAME	Sì	Sì		Sì			Sì	Sì	
<u>Tipo MCA</u>	MCATYPE	Sì	Sì		Sì			Sì	Sì	
<u>Utente agente canale dei messaggi</u>	MCAUSER	Sì	Sì	Sì	Sì		Sì	Sì	Sì	Sì
<u>Nome modo LU 6.2</u>	MODENAME	Sì	Sì		Sì	Sì		Sì	Sì	
<u>monitoraggio</u>	MONCHL	Sì	Sì	Sì	Sì		Sì	Sì	Sì	
<u>Dati utente uscita tentativi messaggio</u>	MRDATA			Sì	Sì				Sì	

Tabella 54. Attributi del canale per i tipi di canali (Continua)




Campo attributo	Parametro del comando MQSC	SDR	SVR	RCV R	RQST R	CONN CLNT	SVR - CONN	CLUS - SDR	CLUS - RCVR	AM QP
<u>Nome uscita nuovo tentativo messaggio</u>	MREXIT			Sì	Sì				Sì	
<u>Conteggio nuovo tentativo di messaggio</u>	MRRTY			Sì	Sì				Sì	
<u>Intervallo nuovo tentativo di messaggio</u>	MRTMR			Sì	Sì				Sì	
<u>Dati utente uscita messaggi</u>	MSGDATA	Sì	Sì	Sì	Sì			Sì	Sì	
<u>Nome uscita messaggi</u>	MSGEXIT	Sì	Sì	Sì	Sì			Sì	Sì	
<u>Priorità di connessione di rete</u>	NETPRTY								Sì	
<u>Velocità messaggio non persistente</u>	NPMSPEED	Sì	Sì	Sì	Sì			Sì	Sì	
<u>Password</u>	Password	Sì	Sì		Sì	Sì		Sì		
<u>numero porta</u>	PORTA									Sì
PROPCTL opzioni di canale per MQGMO	PROPCTL	Sì	Sì					Sì	Sì	
<u>Autorizzazioni Put</u>	PUTAUT			Sì	Sì		 z/OS Sì  Multi No		Sì	
<u>Nome del gestore code</u>	QMNAME					Sì				
 <u>Disposizione</u>	QSGDISP	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	
<u>Dati utente di uscita di ricezione</u>	RCVDATA	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	
<u>Nome utente uscita di ricezione</u>	RCVEXIT	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	
<u>Dati utente di uscita di sicurezza</u>	SCYDATA	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	
<u>Nome uscita di sicurezza</u>	SCYEXIT	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	

Tabella 54. Attributi del canale per i tipi di canali (Continua)

Campo attributo	Parametro del comando MQSC	SDR	SVR	RCV R	RQST R	CONN CLNT	SVR - CONN	CLUS - SDR	CLUS - RCVR	AM QP
<u>Dati utente di uscita di invio</u>	SENDDATA	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	
<u>Nome uscita di invio</u>	SENDEXIT	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	
<u>Wrap numero di sequenza</u>	SEQWRAP	Sì	Sì	Sì	Sì			Sì	Sì	
<u>Connessioni condivise</u>	SHARECNV					Sì	Sì			
<u>Numero di tentativi brevi</u>	SHORTRTY	Sì	Sì					Sì	Sì	
<u>Intervallo nuovo tentativo breve</u>	SHORTTMR	Sì	Sì					Sì	Sì	
 <u>z/OS</u> <u>Protezione dei criteri di protezione</u>	SPLPROT	Sì	Sì	Sì	Sì					
<u>Autenticazione client SSL</u>	SSLCAUTH		Sì	Sì	Sì		Sì		Sì	Sì
<u>Specifica cifratura SSL</u>	SSLCIPH	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì
<u>Peer SSL</u>	SSLPEER	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì
<u>Statistiche di canale</u>	STATCHL	Sì	Sì	Sì	Sì			Sì	Sì	
<u>Nome programma di transazione LU 6.2</u>	TPNAME	Sì	Sì		Sì	Sì		Sì	Sì	
<u>Root argomento</u>	TPROOT									Sì
<u>Tipo di trasporto</u>	TRPTYPE	Sì	Sì	Sì	Sì	Sì	Sì	Sì	Sì	
<u>Utilizza ID client</u>	USECLTID									Sì
<u>Utilizza coda di messaggi non instradabili</u>	USEDLQ	Sì	Sì	Sì	Sì			Sì	Sì	
<u>ID utente</u>	USERID	Sì	Sì		Sì	Sì		Sì		
<u>Nome coda di trasmissione</u>	XMITQ	Sì	Sì							

Tabella 54. Attributi del canale per i tipi di canali (Continua)

Campo attributo	Parametro del comando MQSC	SDR	SVR	RcvR	RQST R	CONN CLNT	SVR - CONN	CLUS - SDR	CLUS - RCVR	AM QP
Note:										
1. Nessuna delle interfacce di gestione consente a questo attributo di essere interrogato o impostato per i canali CLUSSDR. Riceverai un messaggio MQRCCF_WRONG_CHANNEL_TYPE . Tuttavia, l'attributo è presente negli oggetti del canale CLUSSDR (incluse le strutture MQCD) e un'uscita CHAD può impostarlo in modo programmatico, se necessario.										

IBM MQ per alcune piattaforme potrebbe non implementare tutti gli attributi mostrati in questa sezione. Le eccezioni e le differenze della piattaforma sono menzionate nelle descrizioni dei singoli attributi, se pertinenti.

Il nome di ogni attributo viene visualizzato tra parentesi.

Gli attributi sono disposti in ordine alfabetico in gruppi.

Riferimenti correlati

[Comandi MQSC](#)

[MODIFICA CANALE](#)

[Definire il canale](#)

Attributi canale per parole chiave MQSC (A-B)

Un elenco alfabetico degli attributi del canale per le parole chiave MQSC, che inizia con la lettera *A* o *B*.

AFFINITY (affinità di connessione)

Questo attributo specifica se le applicazioni client che si connettono più volte utilizzando lo stesso nome gestore code, utilizzano lo stesso canale client.

Utilizzare questo attributo (MQIACH_CONNECTION_AFFINITY) quando sono disponibili più definizioni di canale applicabili.

I valori possibili sono:

Preferito

La prima connessione in un processo che legge una tabella CCDT (client channel definition table) provvede alla creazione di un elenco di definizioni applicabili basate sul peso del canale client, in cui tutte le definizioni hanno un peso iniziale di 0 e sono disposte in ordine alfabetico. Ciascuna connessione del processo esegue tentativi di connessione utilizzando la prima definizione nell'elenco. Se una connessione non riesce, verrà utilizzata la definizione successiva. Le definizioni non riuscite con valori di peso del canale client diversi 0 vengono spostate in fondo all'elenco. Le definizioni con un peso del canale client pari a 0 rimangono all'inizio dell'elenco e vengono selezionate per prime, per ciascuna connessione.

Ogni processo client con lo stesso nome host crea sempre lo stesso elenco.

Per le applicazioni client scritte in C, C++ o il framework di programmazione .NET (incluso .NETcompletamente gestito) e per le applicazioni che utilizzano IBM MQ classes for Java e IBM MQ classes for JMS, l'elenco viene aggiornato se CCDT è stato modificato dopo la creazione dell'elenco.

Questo valore è quello predefinito e ha il valore 1.

Nessuna

La prima connessione in un processo che legge una tabella CCDT (client channel definition table) provvede alla creazione di un elenco di definizioni applicabili. Tutte le connessioni in un processo

selezionano una definizione applicabile basata sul peso del canale client, in cui tutte le definizioni con un peso pari a 0 vengono selezionate inizialmente in ordine alfabetico.

Per le applicazioni client scritte in C, C++ o il framework di programmazione .NET (incluso .NET completamente gestito) e per le applicazioni che utilizzano IBM MQ classes for Java e IBM MQ classes for JMS, l'elenco viene aggiornato se CCDT è stato modificato dopo la creazione dell'elenco.

Questo attributo è valido solo per il tipo di canale di connessione client.

ALTDATA (Modifica data)

Questo attributo è la data dell'ultima modifica della definizione, nel formato yyyy-mm-dd ed è valido per tutti i tipi di canale.

ALTTIME (ora di modifica)

Questo attributo rappresenta l'ora dell'ultima modifica della definizione, nel formato hh.mm.ss ed è valido per tutti i tipi di canale.

AMQPKA (keep alive AMQP)



Utilizzare l'attributo **AMQPKA** per specificare un tempo keep alive per la connessione del client AMQP. Se il client AMQP non ha inviato alcun frame nell'intervallo keep alive, la connessione viene chiusa.

L'attributo **AMQPKA** determina il valore dell'attributo idle - timeout inviato da IBM MQ a un client AMQP. L'attributo è un periodo di tempo in millisecondi.

Se **AMQPKA** è impostato su un valore > 0, IBM MQ fluisce la metà di tale valore come attributo idle - timeout. Ad esempio, un valore di 10000 fa in modo che il gestore code invii un valore di timeout di inattività di 5000. Il client deve garantire che i dati vengano inviati a IBM MQ almeno ogni 10000 millisecondi. Se i dati non vengono ricevuti da IBM MQ in quel momento, IBM MQ presume che il client abbia perso la connessione e chiude forzatamente la connessione con una condizione di errore `amq:resource-limit-exceeded`.

Un valore di AUTO o 0 indica che IBM MQ non invia un attributo idle - timeout al client AMQP.

Un client AMQP può ancora far fluire un proprio valore di timeout di inattività. In tal caso, IBM MQ invia i dati (o un frame AMQP vuoto) almeno così frequentemente per informare il client che è disponibile.

BATCHHB (Intervallo heartbeat batch)

Il suo attributo consente a un canale di invio di verificare che il canale di ricezione sia ancora attivo prima di eseguire il commit di un batch di messaggi.

L'intervallo di heartbeat batch consente quindi di eseguire il backout del batch piuttosto che diventare in dubbio se il canale ricevente non è attivo. Eseguendo il backout del batch, i messaggi rimangono disponibili per l'elaborazione in modo che possano, ad esempio, essere reindirizzati ad un altro canale.

Se il canale di invio ha avuto una comunicazione dal canale di ricezione entro l'intervallo di heartbeat batch, si presume che il canale di ricezione sia ancora attivo, altrimenti un 'heartbeat' viene inviato al canale di ricezione per il controllo. Il canale di invio attende una risposta dal lato del ricevente del canale per un intervallo, basato sul numero di secondi specificato nell'attributo intervallo heartbeat canale (HBINT).

Il valore è espresso in millisecondi e deve essere compreso nell'intervallo tra zero e 999999. Un valore zero indica che l'heartbeat batch non viene utilizzato.

Questo attributo è valido per i tipi di canale:

- Mittente

- Server
- Mittente del cluster
- Ricevente del cluster

BATCHINT (intervallo batch)

Questo attributo è un periodo, in millisecondi, durante il quale il canale mantiene un batch aperto anche se non vi sono messaggi nella coda di trasmissione.

È possibile specificare qualsiasi numero di millisecondi, da zero a 999 999 999. Il valore predefinito è zero.

Se non si specifica un intervallo batch, il batch viene chiuso quando si verifica una delle seguenti condizioni:

- È stato inviato il numero di messaggi specificato in BATCHSZ.
- È stato inviato il numero di byte specificato in BATCHLIM.
- La coda di trasmissione è vuota.

Sui canali con un carico leggero, dove la coda di trasmissione diventa spesso vuota, la dimensione effettiva del batch potrebbe essere molto più piccola di BATCHSZ.

È possibile utilizzare l'attributo BATCHINT per rendere più efficienti i canali riducendo il numero di batch brevi. Tenere presente, tuttavia, che è possibile rallentare il tempo di risposta, poiché i batch durano più a lungo e i messaggi non vengono sottoposti a commit più a lungo.

Se si specifica BATCHINT, i batch vengono chiusi solo quando viene soddisfatta una delle seguenti condizioni:

- È stato inviato il numero di messaggi specificato in BATCHSZ.
- È stato inviato il numero di byte specificato in BATCHLIM.
- Non vi sono più messaggi nella coda di trasmissione e un intervallo di tempo di BATCHINT è trascorso durante l'attesa dei messaggi (dal momento in cui è stato richiamato il primo messaggio del batch).

Nota: BATCHINT specifica il tempo totale trascorso in attesa di messaggi. Non include il tempo impiegato per richiamare i messaggi che sono già disponibili nella coda di trasmissione o il tempo impiegato per trasferire i messaggi.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Mittente del cluster
- Ricevente del cluster

BATCHLIM (limite batch)

Questo attributo è il limite, in kilobyte, della quantità di dati che possono essere inviati attraverso un canale prima di prendere un punto di sincronizzazione.

Un punto di sincronizzazione viene acquisito dopo che il messaggio che ha causato il raggiungimento del limite è stato trasmesso attraverso il canale.

Il valore deve essere compreso tra 0 e 999999. Il valore predefinito è 5000.

Un valore zero in questo attributo indica che non viene applicato alcun limite di dati ai batch su questo canale.

Il batch viene terminato quando viene soddisfatta una delle seguenti condizioni:

- I messaggi BATCHSZ sono stati inviati.
- I byte BATCHLIM sono stati inviati.

- La coda di trasmissione è vuota e BATCHINT è stato ecceduto.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Mittente del cluster
- Ricevente del cluster

BATCHSZ (dimensione batch)

Questo attributo rappresenta il numero massimo di messaggi da inviare prima di un punto di sincronizzazione.

La dimensione batch non influisce sul modo in cui il canale trasferisce i messaggi; i messaggi vengono sempre trasferiti singolarmente, ma ne viene eseguito il commit o il backout come batch.

Per migliorare le prestazioni, è possibile impostare una dimensione batch per definire il numero massimo di messaggi da trasferire tra due *punti di sincronizzazione*. La dimensione batch da utilizzare viene negoziata quando viene avviato un canale e viene utilizzato il valore più basso delle due definizioni di canale. In alcune implementazioni, la dimensione batch viene calcolata dalla più bassa delle due definizioni di canale e dai due valori MAXUMSGS del gestore code. La dimensione effettiva di un batch può essere inferiore; ad esempio, un batch viene completato quando non ci sono messaggi rimasti nella coda di trasmissione o quando l'intervallo batch scade.

Un valore elevato per la dimensione batch aumenta la velocità di trasmissione, ma i tempi di ripristino aumentano perché ci sono più messaggi di cui eseguire il backout e inviare di nuovo. Il valore BATCHSZ predefinito è 50 e si consiglia di provare prima tale valore. È possibile scegliere un valore inferiore per BATCHSZ se le comunicazioni non sono affidabili, rendendo più probabile il ripristino.

La procedura del punto di sincronizzazione ha bisogno di un identificativo LUW (logical unit of work / unità logica di lavoro) univoco da scambiare attraverso il collegamento ogni volta che viene utilizzato un punto di sincronizzazione, per coordinare le procedure di commit batch.

Se la procedura di commit batch sincronizzato viene interrotta, potrebbe verificarsi una situazione *in dubbio*. Le situazioni in dubbio vengono risolte automaticamente all'avvio di un canale messaggi. Se questa risoluzione non ha esito positivo, potrebbe essere necessario un intervento manuale, utilizzando il comando RESOLVE.

Alcune considerazioni quando si seleziona il numero per la dimensione batch:

- Se il numero è troppo grande, la quantità di spazio della coda occupato su entrambe le estremità del collegamento diventa eccessiva. I messaggi occupano spazio nella coda quando non è stato eseguito il commit e non possono essere rimossi dalle code fino a quando non viene eseguito il commit.
- Se è probabile che vi sia un flusso costante di messaggi, è possibile migliorare le prestazioni di un canale aumentando la dimensione del batch perché sono necessari meno flussi di conferma per trasferire la stessa quantità di byte.
- Se le caratteristiche del flusso di messaggi indicano che i messaggi arrivano in modo intermittente, una dimensione batch di 1 con un intervallo di tempo di disconnessione relativamente grande potrebbe fornire prestazioni migliori.
- Il numero può essere compreso tra 1 e 9999.
- Anche se i messaggi non persistenti su un canale veloce non attendono un punto di sincronizzazione, contribuiscono al conteggio della dimensione batch.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Ricevente
- Richiedente

- Mittente del cluster
- Ricevente del cluster

Attributi di canale per parole chiave MQSC (C)

Un elenco alfabetico degli attributi di canale per parole chiave MQSC, che inizia con la lettera C.

CERTLABL (etichetta certificato)

Questo attributo specifica l'etichetta del certificato della definizione di canale.

L'etichetta identifica quale certificato personale nel repository delle chiavi viene inviato al peer remoto. Il certificato è definito come descritto in [Etichette certificato digitale](#).

I canali in entrata (inclusi i canali RCVR, RQSTR, CLUSRCVR, unqualified SERVER e SVRCONN) inviano il certificato configurato solo se la versione IBM MQ del peer remoto supporta completamente la configurazione dell'etichetta del certificato e il canale utilizza una CipherSpecTLS.

In caso contrario, l'attributo **CERTLABL** del gestore code determina il certificato inviato. Questa limitazione è dovuta al fatto che il meccanismo di selezione dell'etichetta del certificato per i canali in entrata dipende da un'estensione del protocollo TLS che non è supportata in tutti i casi. In particolare, i client Java e JMS non supportano l'estensione del protocollo richiesta e riceveranno solo il certificato configurato dall'attributo **CERTLABL** del gestore code, indipendentemente dall'impostazione dell'etichetta specifica del canale.

Un canale server non qualificato è uno che non ha il campo CONNAME impostato.

Nessuna delle interfacce di gestione consente a questo attributo di essere interrogato o impostato per i canali CLUSSDR. Si riceverà un messaggio MQRCCF_WRONG_CHANNEL_TYPE. Tuttavia, l'attributo è presente negli oggetti del canale CLUSSDR (include le strutture MQCD) e un'uscita CHAD può impostarlo in modo programmatico, se necessario.

Per ulteriori informazioni su cosa può contenere l'etichetta del certificato, consultare [Etichette del certificato digitale, comprensione dei requisiti](#).

Questo attributo è valido per tutti i tipi di canale.

Nota: Per SSL/TLS, CERTLABL deve essere definito nella definizione QMGR. È possibile, facoltativamente, impostare un CERTLABL sulla definizione CHANNEL.

Il gestore code CERTLABL è selezionato e deve essere un certificato personale valido, anche se si sta impostando CERTLABL sulla definizione CHANNEL.

CHANNEL (Nome canale)

Questo attributo specifica il nome della definizione di canale.

Il nome può contenere fino a 20 caratteri, anche se poiché entrambe le estremità di un canale di messaggi devono avere lo stesso nome e altre implementazioni potrebbero avere limitazioni sulla dimensione, il numero effettivo di caratteri potrebbe essere più piccolo.

Dove possibile, i nomi dei canali sono univoci per un canale tra due gestori code in una rete di gestori code interconnessi.

Il nome deve contenere i caratteri del seguente elenco:

Alfabetico	(A-Z, a - z; notare che maiuscole e minuscole sono significative)
Numeriche	(0-9)
Periodo	(.)
barra	(/)
Carattere di sottolineatura	(_)

Segno percentuale (%)

Nota:

1. Gli spazi incorporati non sono consentiti e gli spazi iniziali vengono ignorati.
2. Sui sistemi che utilizzano EBCDIC Katakana, è impossibile utilizzare caratteri minuscoli.

Questo attributo è valido per tutti i tipi di canale.

CHLTYPE (Tipo di canale)

Questo attributo specifica il tipo di canale definito.


I tipi di canale possibili sono:

Tipi di canale di messaggi:

- Mittente
- Server
- Ricevente
- Richiedente
- Mittente del cluster
- Ricevente del cluster

Tipi di canale MQI:

- Connessione client (solo AIX, Linux, and Windows)

Nota:  I canali di connessione client possono essere definiti anche su z/OS per essere utilizzati su altre piattaforme.

- Connessione server
- AMQP

Le due estremità di un canale devono avere lo stesso nome e tipi compatibili:

- Mittente con destinatario
- Richiedente con server
- Richiedente con mittente (per callback)
- Server con destinatario (il server viene utilizzato come mittente)
- Connessione client con connessione server
- Mittente cluster con ricevente cluster
- AMQP con AMQP

CLNTWGHT (Peso canale client)

Questo attributo specifica un peso per influenzare quale definizione di canale di connessione client viene utilizzata.

L'attributo del peso del canale client viene utilizzato in modo che sia possibile selezionare le definizioni di canale client casualmente, sulla base del loro peso, nel caso in cui sia disponibile più di una definizione adatta.

Quando un client emette una connessione MQCONN che richiede una connessione a un gruppo di gestori code, specificando un nome gestore code che inizia con un asterisco, che abilita il bilanciamento del peso del client su diversi gestori code, e più di una definizione di canale adatta è disponibile nella tabella di definizione del canale client (CCDT), la definizione da utilizzare viene selezionata in modo casuale in base al peso, con tutte le definizioni CLNTWGHT (0) applicabili selezionate per prime in ordine alfabetico.

Nota: Quando si utilizza una CCDT JSON è possibile avere più canali con lo stesso nome. Se esistono più canali con lo stesso nome e hanno CLNTWGHT (0), i canali verranno selezionati nell'ordine in cui sono definiti nella CCDT JSON.

Specificare un valore compreso nell'intervallo 0 - 99. Il valore predefinito è 0.

Il valore 0 indica che non viene eseguito alcun bilanciamento del carico e che le funzioni applicabili vengono selezionate in ordine alfabetico. Per abilitare il bilanciamento del carico, scegliere un valore compreso fra 1 e 99, dove 1 è il peso minore e 99 quello maggiore. La distribuzione delle connessioni tra due o più canali con pesi diversi da zero è proporzionale al rapporto di tali pesi. Ad esempio, tre canali con valori CLNTWGHT di 2, 4 e 14 vengono selezionati circa il 10%, 20% e 70% del tempo. Questa distribuzione non è garantita. Se l'attributo AFFINITY della connessione è impostato su PREFERRED, la prima connessione sceglie una definizione di canale in base alle ponderazioni del client e quindi le connessioni successive continuano a utilizzare la stessa definizione di canale.

Questo attributo è valido solo per il tipo di canale di connessione client.

CLUSNL (Elenco nomi cluster)

Questo attributo è il nome dell'elenco nomi che specifica un elenco di cluster a cui appartiene il canale.

Fino a uno dei valori risultanti di CLUSTER o CLUSNL può essere non vuoto. Se uno dei valori non è vuoto, l'altro deve essere vuoto.

Questo attributo è valido per i tipi di canale:

- Mittente del cluster
- Ricevente del cluster

Cluster (CLUSTER)

Questo attributo è il nome del cluster a cui appartiene il canale.

La lunghezza massima è di 48 caratteri conformi alle regole per la denominazione degli oggetti IBM MQ .

Fino a uno dei valori risultanti di CLUSTER o CLUSNL può essere non vuoto. Se uno dei valori non è vuoto, l'altro deve essere vuoto.

Questo attributo è valido solo per i tipi di canale:

- Mittente del cluster
- Ricevente del cluster

CLWLPRTY (Priorità carico di lavoro cluster)

L'attributo del canale CLWLPRTY specifica l'ordine di priorità per i canali per la distribuzione del workload del cluster. Il valore deve essere compreso nell'intervallo 0-9, dove 0 è la priorità più bassa e 9 è la più alta.

Utilizzare l'attributo del canale CLWLPRTY per impostare un ordine di priorità per le destinazioni cluster disponibili. IBM MQ seleziona le destinazioni con la priorità più alta prima di selezionare le destinazioni con la priorità di destinazione cluster più bassa. Se ci sono più destinazioni con la stessa priorità, seleziona la destinazione utilizzata meno di recente.

Se vi sono due destinazioni possibili, è possibile utilizzare questo attributo per consentire il failover. I messaggi vengono inviati al gestore code con il canale con priorità più alta. Se diventa non disponibile, i messaggi passano al successivo gestore code con priorità più alta. I gestori code con priorità più bassa agiscono come riserve.

IBM MQ controlla lo stato del canale prima di assegnare la priorità ai canali. Solo i gestori code sono candidati per la selezione.

Note:

- Specificare questo attributo sul canale ricevente del cluster sul gestore code di destinazione. È probabile che qualsiasi bilanciamento specificato sul canale mittente del cluster corrispondente venga ignorato. Consultare [Canali cluster](#).
- La disponibilità di un gestore code remoto si basa sullo stato del canale per tale gestore code. Quando i canali vengono avviati, il loro stato cambia diverse volte, con alcuni degli stati meno preferenziali rispetto all'algoritmo di gestione del workload del cluster. In pratica ciò significa che è possibile scegliere destinazioni con priorità più bassa (backup) mentre i canali verso destinazioni con priorità più alta (primaria) sono in fase di avvio.
- Se è necessario assicurarsi che nessun messaggio vada a una destinazione di backup, non utilizzare CLWLPRTY. Prendere in considerazione l'utilizzo di code separate o di CLWLRANK con un passaggio manuale dal primario al backup.

CLWLRANK (classificazione carico di lavoro cluster)

L'attributo del canale **CLWLRANK** specifica la classificazione dei canali per la distribuzione del workload del cluster. Il valore deve essere compreso nell'intervallo 0-9, dove 0 è il rango più basso e 9 è il più alto.

Utilizzare l'attributo del canale **CLWLRANK** se si desidera controllare la destinazione finale per i messaggi inviati a un gestore code in un altro cluster. Controllare la scelta della destinazione finale impostando la classificazione dei canali che collegano un gestore code ai gestori code gateway all'intersezione dei cluster.

Quando si imposta **CLWLRANK**, i messaggi prendono un instradamento specificato attraverso i cluster interconnessi verso una destinazione di livello superiore. Ad esempio, i messaggi arrivano a un gestore code del gateway che può inviarli a uno dei due gestori code utilizzando i canali 1 e 2. Vengono inviati automaticamente al gestore code connesso da un canale con la classificazione più elevata, in questo caso il canale al gestore code classificato 2.

IBM MQ ottiene la classificazione dei canali prima di controllarne lo stato. Ottenere la classificazione prima di controllare lo stato del canale significa che anche i canali non accessibili sono disponibili per la selezione. Consente l'instradamento dei messaggi attraverso la rete anche se la destinazione finale non è disponibile.

Note:

- Specificare questo attributo sul canale ricevente del cluster sul gestore code di destinazione. È probabile che qualsiasi bilanciamento specificato sul canale mittente del cluster corrispondente venga ignorato. Consultare [Canali cluster](#).
- Se è stato utilizzato anche l'attributo di priorità **CLWLPRTY**, IBM MQ seleziona tra le destinazioni disponibili. Se un canale non è disponibile per la destinazione con la classificazione più alta, il messaggio viene conservato nella coda di trasmissione. Viene rilasciato quando il canale diventa disponibile. Il messaggio non viene inviato alla successiva destinazione disponibile nell'ordine di classificazione.

CLWLWGHT (Peso carico di lavoro cluster)

L'attributo di canale CLWLWGHT specifica il peso applicato ai canali CLUSSDR e CLUSRCVR per la distribuzione del carico di lavoro del cluster. Il valore deve essere compreso tra 1 e 99, dove 1 è il peso più basso e 99 è il più alto.

Utilizzare CLWLWGHT per inviare più messaggi ai server con maggiore potenza di elaborazione. Maggiore è il peso del canale, maggiore è il numero di messaggi inviati su tale canale.

Note:

- Specificare questo attributo sul canale ricevente del cluster sul gestore code di destinazione. È probabile che qualsiasi bilanciamento specificato sul canale mittente del cluster corrispondente venga ignorato. Consultare [Canali cluster](#).
- Quando CLWLWGHT viene modificato dal valore predefinito di 50 su qualsiasi canale, il bilanciamento del carico di lavoro dipende dal numero totale di volte in cui ogni canale è stato scelto per un messaggio

inviato a una coda cluster. Per ulteriori informazioni, consultare [“L'algoritmo di gestione del workload del cluster”](#) a pagina 147.

COMPHDR (compressione intestazione)

Questo attributo è un elenco di tecniche di compressione dei dati di intestazione supportate dal canale.

Per i canali mittente, server, mittente cluster, destinatario cluster e connessione client, i valori specificati sono in ordine di preferenza con la prima tecnica di compressione supportata dall'estremità remota del canale utilizzato. Le tecniche di compressione reciprocamente supportate dei canali vengono trasmesse all'uscita del messaggio del canale di invio in cui la tecnica di compressione utilizzata può essere modificata in base al messaggio. La compressione modifica i dati passati per inviare e ricevere uscite.

I possibili valori sono:

Nessuna

Nessuna compressione dati di intestazione eseguita. Questo è il valore predefinito.

SISTEMA

Compressione dati di intestazione eseguita correttamente.

Questo attributo è valido per tutti i tipi di canale.

COMPMSG (Compressione dati)

Questo attributo è un elenco di tecniche di compressione dei dati dei messaggi supportate dal canale.

Per i canali mittente, server, mittente cluster, ricevente cluster e connessione client, i valori specificati sono in ordine di preferenza. Viene utilizzata la prima tecnica di compressione supportata dall'estremità remota del canale. Le tecniche di compressione reciprocamente supportate dei canali vengono trasmesse all'uscita del messaggio del canale di invio in cui la tecnica di compressione utilizzata può essere modificata in base al messaggio. La compressione modifica i dati passati per inviare e ricevere uscite. Consultare [“COMPHDR \(compressione intestazione\)”](#) a pagina 101 per la compressione dell'intestazione del messaggio.

I valori possibili sono:

Nessuna

Nessuna compressione dati di messaggi eseguita. Questo è il valore predefinito.

RLE

La compressione dei dati dei messaggi è stata eseguita mediante la codifica run-length.

ZLIBFAST

La compressione dei dati dei messaggi è stata eseguita mediante la tecnica di compressione zlib. È preferibile che il tempo di compressione sia breve.

ZLIBFAST può essere scaricato facoltativamente nella funzione di compressione dati zEnterprise . Per ulteriori informazioni, consultare [zEDC Express facility](#) .

ZLIBHIGH

La compressione dei dati dei messaggi è stata eseguita mediante la tecnica di compressione zlib. È preferibile che il tempo di compressione sia elevato.

V 9.4.0 LZ4FAST

La compressione dei dati dei messaggi viene eseguita utilizzando la tecnica di compressione LZ4. È preferibile che il tempo di compressione sia breve.

V 9.4.0 LZ4HIGH

La compressione dei dati dei messaggi viene eseguita utilizzando la tecnica di compressione LZ4 . È preferibile che il tempo di compressione sia elevato.

ANY

Consente al canale di supportare qualsiasi tecnica di compressione supportata dal gestore code. Supportato solo per i canali Receiver, Requester e Server - Connection.

Questo attributo è valido per tutti i tipi di canale.

AIX Da IBM MQ 9.3.0, le tecniche ZLIBFAST e ZLIBHIGH possono utilizzare la libreria zlibNX accelerata dall'hardware su IBM MQ for AIX se è installata. La libreria zlibNX è una versione migliorata della libreria di compressione zlib che supporta la compressione e la decompressione dei dati accelerata dall'hardware utilizzando i coprocessori denominati Nest accelerators (NX) su server basati su processore IBM POWER9 . La libreria zlibNX è disponibile in IBM AIX 7.2 con Technology Level 4 Expansion Pack e versioni successive. I messaggi altamente comprimibili che superano la dimensione di 2KB hanno maggiori probabilità di trarre vantaggio dalla scelta di utilizzare la libreria zlibNX , riducendo l'utilizzo della CPU. Per abilitare un MCA (message channel agent) all'utilizzo della libreria zlibNX , impostare la variabile di ambiente AMQ_USE_ZLIBNX.

CONNNAME (Nome connessione)

Questo attributo è l'identificativo della connessione di comunicazione. Specifica i particolari collegamenti di comunicazione che devono essere utilizzati da questo canale.

È facoltativo per i canali server, a meno che il canale server non venga attivato, nel qual caso è necessario specificare un nome di connessione.

Specificare **CONNNAME** come un elenco separato da virgole di nomi di macchine per il **TRPTYPE** indicato. In genere, è richiesto un solo nome macchina. È possibile fornire più nomi macchina al fine di configurare più connessioni con le stesse proprietà. Le connessioni di solito vengono tentate nell'ordine in cui sono specificate nell'elenco, finché non ne viene stabilita una. L'ordine viene modificato per i client se viene fornito l'attributo **CLNTWGHT**. Se nessuna connessione ha esito positivo, il canale tenta di nuovo la connessione come determinato dagli attributi del canale. Con i canali del client, un elenco di connessioni fornisce un'alternativa all'utilizzo dei gruppi di gestori code per configurare più connessioni. Con i canali dei messaggi, un elenco di connessioni viene utilizzato per configurare le connessioni agli indirizzi alternativi di un gestore code a più istanze.

Multi Su Multiplatforme, il parametro del nome connessione TCP/IP di un canale ricevente del cluster è facoltativo. Se si lascia vuoto il nome connessione, IBM MQ genera un nome connessione per conto dell'utente, presumendo la porta predefinita e utilizzando l'indirizzo IP corrente del sistema. È possibile sovrascrivere il numero di porta predefinito, utilizzando comunque l'indirizzo IP corrente del sistema. Per ciascun nome di connessione, lasciare vuoto il nome IP e fornire il numero di porta tra parentesi; ad esempio:

(1415)

Il **CONNNAME** generato è sempre nel formato decimale puntato (IPv4) o esadecimale (IPv6), invece che nel formato di nome host DNS alfanumerico.

La lunghezza massima del nome dipende dalla piattaforma:

- **Multi** 264 caratteri.
- **z/OS** 48 caratteri (vedere [nota 1](#)).

Se il tipo di trasporto è TCP

CONNNAME è il nome host o l'indirizzo di rete della macchina remota (o la macchina locale per i canali riceventi del cluster). Ad esempio, (ABC.EXAMPLE.COM), (2001:DB8:0:0:0:0:0:0) o (127.0.0.1). Può contenere il numero di porta, ad esempio (MACHINE(123)).

z/OS Può includere il nome IP di un gruppo DNS dinamico o una porta di input Network Dispatcher .

Se si utilizza un indirizzo IPv6 in una rete che supporta solo IPv4, il nome della connessione non viene risolto. In una rete che utilizza sia IPv4 che IPv6, il nome della connessione interagisce con l'indirizzo locale per stabilire quale stack IP viene utilizzato. Consultare [“LOCLADDR \(Indirizzo locale\)”](#) a [pagina 107](#) per ulteriori informazioni.

Se il tipo di trasporto è LU 6.2

Muti Se vengono specificati TPNAME e MODENAME, fornire il nome completo della LU partner. Se TPNAME e MODENAME sono vuoti, fornire il nome oggetto delle informazioni lato CPI-C per la piattaforma specifica.

z/OS Esistono due forme in cui specificare il valore:

- Nome unità logica

Le informazioni sull'unità logica per il gestore code, che comprendono il nome dell'unità logica, il nome TP e il nome della modalità facoltativa. Questo nome può essere specificato in uno dei tre formati:

<i>Tabella 55. Nomi e formati di unità logiche</i>	
Formato	Esempio
luname	IGY12355
luname/TPname	IGY12345/APING
luname/TPname/modename	IGY12345/APINGD/#INTER

Per il primo modulo, il nome TP e il nome modalità devono essere specificati per gli attributi TPNAME e MODENAME; altrimenti questi attributi devono essere vuoti. Per i canali di connessione client, è consentito solo il primo modulo.

- Nome simbolico

Il nome di destinazione simbolico per le informazioni sull'unità logica per il gestore code, come definito nel dataset delle informazioni lato. Gli attributi TPNAME e MODENAME devono essere vuoti. Per i canali riceventi del cluster, le informazioni laterali si trovano sugli altri gestori code del cluster. In questo caso, può essere un nome che un'uscita di definizione automatica del canale può risolvere nelle informazioni sull'unità logica appropriate per il gestore code locale.

Il nome LU specificato o implicito può essere quello di un gruppo di risorse generico VTAM.

Se il protocollo di trasmissione è NetBIOS

CONNNAME è il nome NetBIOS definito sulla macchina remota.

Se il protocollo di trasmissione è SPX

CONNNAME è un indirizzo in stile SPX composto da un indirizzo di rete a 4 byte, un indirizzo di nodo a 6 byte e un numero socket a 2 byte. Immettere questi valori in formato esadecimale, con gli indirizzi di rete e nodo separati da un punto e il numero socket tra parentesi. Ad esempio:

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

Se il numero di socket viene omissso, viene utilizzato il numero di socket SPX IBM MQ predefinito. Il valore predefinito è X'5E86'.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server

È facoltativo per i canali server, a meno che il canale server non venga attivato, nel qual caso è necessario specificare un nome di connessione.

- Richiedente
- Connessione client
- Mittente del cluster
- Ricevente del cluster

Nota:

1. Per la lunghezza dei nomi, è possibile aggirare il limite di 48 caratteri in uno dei modi seguenti:
 - Imposta i tuoi server DNS in modo da utilizzare, ad esempio, il nome host "myserver" invece di "myserver.location.company.com", assicurandoti di poter utilizzare il nome host breve.
 - Utilizzare gli indirizzi IP.
2. La definizione del protocollo di trasmissione è contenuta in [TRPTYPE](#).

CONVERT (Conversione messaggio)

Questo attributo specifica che il messaggio deve essere convertito nel formato richiesto dal sistema ricevente prima della trasmissione.

I dati del messaggio dell'applicazione vengono generalmente convertiti dall'applicazione ricevente. Tuttavia, se il gestore code remoto si trova su una piattaforma che non supporta la conversione dei dati, utilizzare questo attributo del canale per specificare che il messaggio deve essere convertito nel formato richiesto dal sistema ricevente **prima della trasmissione**.

I valori possibili sono yes e no. Se si specifica yes, i dati dell'applicazione nel messaggio vengono convertiti prima dell'invio se è stato specificato uno dei nomi di formato integrati oppure se è disponibile un'uscita di conversione dati per un formato definito dall'utente (consultare [Scrittura delle uscite di conversione dati](#)). Se si specifica no, i dati dell'applicazione nel messaggio non vengono convertiti prima dell'invio.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Mittente del cluster
- Ricevente del cluster

Attributi del canale per le parole chiave MQSC (D-L)

Un elenco alfabetico degli attributi di canale per le parole chiave MQSC, che inizia con le lettere *D* thru *L*.

DEFRECON (riconnesione predefinita)

Specifica se una connessione client riconnette automaticamente un'applicazione client in caso di interruzione della connessione.

I valori possibili sono:

NO (valore predefinito)

A meno che non venga sovrascritto da **MQCONN**, il client non viene riconnesso automaticamente.

Sì

A meno che non venga sovrascritta da **MQCONN**, il client si riconnette automaticamente.

QMGR

A meno che non venga sovrascritto da **MQCONN**, il client si riconnette automaticamente, ma solo allo stesso gestore code. L'opzione QMGR ha lo stesso effetto di MQCNO_RECONNECT_Q_MGR.

Disabilitato

La riconnesione è disabilitata, anche se richiesta dal programma client utilizzando la chiamata MQI **MQCONN**.

Questo attributo è valido solo per canali di connessione client.

DESCR (Descrizione)

Questo attributo descrive la definizione del canale e contiene fino a 64 byte di testo.

Nota: Il numero massimo di caratteri è ridotto se il sistema utilizza caratteri DBCS (double byte character set).

Utilizzare i caratteri della serie di caratteri identificati dal CCSID (Coded Character Set Identifier) per il gestore code per garantire che il testo venga convertito correttamente se viene inviato a un altro gestore code.

Questo attributo è valido per tutti i tipi di canale.

DISCINT (Intervallo di disconnessione)

Questo attributo è il periodo di tempo dopo il quale un canale viene chiuso, se non arriva alcun messaggio durante tale periodo.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Connessione server che utilizza solo il protocollo TCP
- Mittente del cluster
- Ricevente del cluster

Questo attributo è un timeout, specificato in secondi.

Per i canali di messaggi server - to - server (i tipi di canale server, mittente del cluster, mittente e ricevente del cluster), l'intervallo viene misurato dal punto in cui termina un batch, ossia quando viene raggiunta la dimensione del batch o quando l'intervallo batch scade e la coda di trasmissione diventa vuota. Se nessun messaggio arriva sulla coda di trasmissione durante l'intervallo di tempo specificato, il canale si chiude. (Il tempo è approssimativo.)

Lo scambio di dati di controllo tra le due estremità del canale di messaggi server - to - server include un'indicazione del motivo della chiusura. Ciò garantisce che l'estremità corrispondente del canale rimanga disponibile per il riavvio.

È possibile specificare qualsiasi numero di secondi compreso tra zero e 999 999, dove il valore zero indica nessuna disconnessione; attendere indefinitamente.

Il valore predefinito è 6000 secondi (100 minuti) per i canali di messaggi server - to - server e 0 (nessun timeout) per i canali di collegamento server. È possibile modificare il valore predefinito utilizzato per i nuovi canali creati, modificando gli oggetti canale predefiniti. Ad esempio, modificare l'attributo DISCINT su SYSTEM.DEF.SENDER per fornire un nuovo valore predefinito per i nuovi canali mittente definiti.

Per i canali di connessione server che utilizzano il protocollo TCP, l'intervallo rappresenta il valore di disconnessione inattività del client, specificato in secondi. Se un programma del canale di connessione server non ha ricevuto alcuna comunicazione dal client partner per questo periodo di tempo, termina la connessione.

L'intervallo di inattività della connessione server si applica tra chiamate API IBM MQ da un client.

Nota: Un MQGET con una chiamata di attesa potenzialmente di lunga durata non è classificato come inattività e, pertanto, non va mai in timeout a causa della scadenza di DISCINT.



Attenzione: Le prestazioni sono influenzate dal valore specificato per l'intervallo di disconnessione.

Un valore basso (ad esempio, alcuni secondi) può essere dannoso per le prestazioni del sistema arrestando e riavviando costantemente il canale. Un valore elevato (più di un'ora) potrebbe significare che le risorse di sistema vengono utilizzate senza vantaggi. È anche possibile specificare un intervallo di heartbeat, in modo che quando non ci sono messaggi nella coda di trasmissione, l'MCA mittente invia un flusso di heartbeat all'MCA ricevente, offrendo così all'MCA ricevente l'opportunità di sospendere il canale senza attendere la scadenza dell'intervallo di disconnessione. Affinché questi due valori funzionino insieme in modo efficace, il valore dell'intervallo di heartbeat deve essere significativamente inferiore al valore dell'intervallo di disconnessione.

Il valore DISCINT predefinito per i canali di messaggi server - to - server è 6000 secondi (100 minuti). Tuttavia, un valore di pochi minuti è spesso un valore ragionevole da utilizzare senza influire sulle prestazioni o mantenere i canali in esecuzione per periodi di tempo inutilmente lunghi. Se è appropriato per il proprio ambiente, è possibile modificare questo valore, su ogni singolo canale o modificando l'attributo DISCINT nelle definizioni di canale predefinite (ad esempio, SYSTEM.DEF.SENDER per i canali mittente) prima di creare i propri canali.

Per ulteriori informazioni, vedi [Arresto e disattivazione dei canali](#).

HBINT (Intervallo heartbeat)

Questo attributo specifica il tempo approssimativo tra i flussi di heartbeat che devono essere passati da un MCA (message channel agent) di invio quando non ci sono messaggi nella coda di trasmissione.

I flussi di heartbeat sbloccano l'MCA ricevente, che è in attesa dell'arrivo dei messaggi o della scadenza dell'intervallo di disconnessione. Quando l'MCA ricevente viene sbloccato, può disconnettere il canale senza attendere la scadenza dell'intervallo di disconnessione. I flussi heartbeat, inoltre, liberano i buffer di memoria assegnati per i messaggi di grandi dimensioni e chiudono le code che sono state lasciate aperte all'estremità di ricezione del canale.

Il valore è espresso in secondi e deve essere compreso tra 0 e 999 999. Il valore zero indica che non devono essere inviati flussi heartbeat. Il valore predefinito è 300. Per essere più utile, il valore deve essere significativamente inferiore al valore dell'intervallo di disconnessione.

Con le applicazioni che utilizzano le API IBM MQ classes for Java, JMS o .NET , il valore HBINT viene determinato in uno dei seguenti modi:

- Dal valore sul canale SVRCONN utilizzato dall'applicazione.
- Oppure in base al valore sul canale CLNTCONN, se l'applicazione è stata configurata per utilizzare una CCDT.

Per i canali di connessione server e client, gli heartbeat possono fluire sia dal lato server che dal lato client in maniera indipendente. Se non è stato trasferito alcun dato attraverso il canale per l'intervallo heartbeat, l'agent MQI di connessione client invia un flusso heartbeat e l'agent MQI di connessione server risponde con un altro flusso heartbeat. Ciò si verifica indipendentemente dallo stato del canale, ad esempio, indipendentemente dal fatto che sia inattivo durante l'esecuzione di una chiamata API o che sia inattivo in attesa dell'input dell'utente client. L'agent MQI di connessione server è anche in grado di inizializzare un heartbeat sul client, sempre indipendentemente dallo stato del canale. Per impedire che gli agent MQI di connessione server e di connessione client si battano reciprocamente nello stesso momento, l'heartbeat del server viene trasmesso dopo che non è stato trasferito alcun dato attraverso il canale per l'intervallo di heartbeat più 5 secondi.

Per i canali di connessione server e client che operano in modalità canale prima di IBM WebSphere MQ 7.0, gli heartbeat vengono trasmessi solo quando un MCA del server è in attesa di un comando MQGET con l'opzione WAIT specificata, che è stata emessa per conto di una applicazione client.


Per ulteriori informazioni relative al funzionamento dei canali MQI nelle due modalità, consultare [SharingConversations \(MQLONG\)](#).

KAINT (intervallo keepalive)

Questo attributo viene utilizzato per specificare un valore di timeout per un canale.

L'attributo Intervallo keepalive è un valore passato allo stack di comunicazione che specifica il tempo Keepalive per il canale. Consente di specificare un diverso valore keepalive per ogni canale.

È possibile impostare l'attributo Intervallo keepalive (KAINT) per canali su base per canale.

 Su Multipiattaforme, è possibile accedere e modificare il parametro, ma viene solo memorizzato e inoltrato; non esiste alcuna implementazione funzionale del parametro. Se è necessaria la funzionalità fornita dal parametro KAJNT, utilizzare il parametro HBINT (Intervallo heartbeat), come descritto in [“HBINT \(Intervallo heartbeat\)”](#) a pagina 106.

Affinché questo attributo abbia effetto, è necessario abilitare TCP/IP keepalive.

- **z/OS** Su z/OS, abilitare keepalive immettendo il comando ALTER QMGR TCPKEEP (YES) MQSC.
- **Multi** In Multiplatforme, si verifica quando il parametro KEEPALIVE=YES viene specificato nella stanza TCP nel file di configurazione dell'accodamento distribuito, `qm.ini`, o tramite IBM MQ Explorer.

Keepalive deve essere abilitato anche all'interno di TCP/IP stesso, utilizzando il dataset di configurazione del profilo TCP.

Il valore indica un'ora, in secondi, e deve essere compreso tra 0 e 99999. Un valore Intervallo keepalive pari a 0 indica che il keepalive specifico del canale non è abilitato per il canale e viene utilizzato solo il valore Keepalive di sistema impostato in TCP/IP. È anche possibile impostare KAIN T su un valore AUTO (questo è il valore predefinito). Se KAIN T è impostato su AUTO, il valore Keepalive si basa sul valore dell'intervallo heartbeat negoziato (HBINT) come segue:

HBINT negoziato	KAIN T
>0	HBINT + 60 secondi negoziati
0	0

Questo attributo è valido per tutti i tipi di canale.

Il valore viene ignorato per tutti i canali che hanno un TransportType (TRPTYPE) diverso da TCP o SPX

LOCLADDR (Indirizzo locale)

Questo attributo specifica l'indirizzo di comunicazione locale per il canale.

Nota: I canali AMQP non supportano lo stesso formato di LOCLADDR degli altri canali IBM MQ . Per ulteriori informazioni, consultare [“#unique_51/unique_51_Connect_42_locladdr_amqp”](#) a pagina 110.

LOCLADDR per tutti i canali tranne i canali AMQP

Questo attributo si applica solo se il tipo di trasporto (TRPTYPE) è TCP/IP. Per tutti gli altri tipi di trasporto, viene ignorato.

Quando viene specificato un valore LOCLADDR , un canale arrestato e quindi riavviato continua ad utilizzare l'indirizzo TCP/IP specificato in LOCLADDR. Negli scenari di ripristino, questo attributo potrebbe essere utile quando il canale comunica attraverso un firewall. È utile perché rimuove i problemi causati dal riavvio del canale con l'indirizzo IP dello stack TCP/IP a cui è connesso. LOCLADDR può anche forzare un canale ad utilizzare uno stack IPv4 o IPv6 su un sistema a doppio stack o uno stack a modalità doppia su un sistema a singolo stack.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Richiedente
- Connessione client
- Mittente del cluster
- Ricevente del cluster

Quando LOCLADDR include un indirizzo di rete, l'indirizzo deve essere un indirizzo di rete appartenente a un'interfaccia di rete sul sistema su cui viene eseguito il canale. Ad esempio, quando si definisce un canale mittente sul gestore code ALPHA al gestore code BETA con il seguente comando MSQC:

```
DEFINE CHANNEL(TO.BETA) CHLTYPE(SDR) CONNAME(192.0.2.0) XMITQ(BETA) LOCLADDR(192.0.2.1)
```

L'indirizzo LOCLADDR è l'IPv4 indirizzo 192.0.2.1. Questo canale mittente viene eseguito sul sistema del gestore code ALPHA, per cui l'indirizzo IPv4 deve appartenere a una delle interfacce di rete del sistema.

Il valore è l'indirizzo IP facoltativo e la porta o l'intervallo di porte facoltativi utilizzati per le comunicazioni TCP/IP in uscita. Il formato di queste informazioni è il seguente:

```
LOCLADDR([ip-addr] [(low-port[,high-port])][, [ip-addr] [(low-port[,high-port])]])
```

La lunghezza massima di **LOCLADDR**, inclusi più indirizzi, è MQ_LOCAL_ADDRESS_LENGTH.

Se si omette **LOCLADDR**, viene automaticamente assegnato un indirizzo locale.

Nota: è possibile impostare **LOCLADDR** per un client C utilizzando la CCDT (Client Channel Definition Table).

Tutti i parametri sono facoltativi. L'omissione della parte `ip-addr` dell'indirizzo è utile per abilitare la configurazione di un numero di porta fisso per un firewall IP. L'omissione del numero di porta è utile per selezionare uno specifico adattatore di rete senza dover identificare un numero di porta locale univoco. Lo stack TCP/IP genera un numero di porta univoco.

Specificare `[, [ip-addr] [(low-port[,high-port])]]` più volte per ogni indirizzo locale aggiuntivo. Utilizzare più indirizzi locali se si desidera specificare uno specifico sottoinsieme di adattatori di rete locali. È anche possibile utilizzare `[, [ip-addr] [(low-port[,high-port])]]` per rappresentare un particolare indirizzo di rete locale su server differenti che fanno parte di una configurazione di gestore code a più istanze.

ip-addr

`ip-addr` è specificato in uno di tre formati:

decimale puntata IPv4

Ad esempio, 192.0.2.1

notazione esadecimale IPv6

Ad esempio, 2001:DB8:0:0:0:0:0:0

Formato nome host alfanumerico

Ad esempio WWW.EXAMPLE.COM

low-port e high-port

`low-port` e `high-port` sono numeri di porta racchiusi tra parentesi.

La seguente tabella mostra come è possibile utilizzare il parametro **LOCLADDR**:

<i>Tabella 57. Esempi di come è possibile utilizzare il parametro LOCLADDR</i>	
LOCLADDR	Significato
9.20.4.98	Il canale si collega localmente a questo indirizzo
9.20.4.98, 9.20.4.99	Il canale si collega all'indirizzo IP. L'indirizzo potrebbe essere due adattatori di rete su un server o un adattatore di rete differente su due server diversi in una configurazione a più istanze.
9.20.4.98(1000)	Il canale si collega localmente a questo indirizzo e alla porta 1000
9.20.4.98(1000,2000)	Il canale si collega localmente a questo indirizzo e alla porta compresa nell'intervallo 1000 - 2000
(1000)	Il canale si collega localmente alla porta 1000

Tabella 57. Esempi di come è possibile utilizzare il parametro **LOCLADDR** (Continua)

LOCLADDR	Significato
(1000,2000)	Il canale si collega localmente alla porta compresa tra 1000 e 2000

Quando viene avviato un canale, i valori specificati per il nome della connessione (CONNAME) e l'indirizzo locale (LOCLADDR) determinano quale stack IP viene utilizzato per le comunicazioni. Lo stack IP utilizzato è determinato come segue:

- Se il sistema ha solo uno stack IPv4 configurato, lo stack IPv4 viene sempre utilizzato. Se un indirizzo locale (LOCLADDR) o un nome connessione (CONNAME) viene specificato come indirizzo di rete IPv6, viene generato un errore e il canale non viene avviato.
- Se il sistema ha solo uno stack IPv6 configurato, lo stack IPv6 viene sempre utilizzato. Se un indirizzo locale (LOCLADDR) viene specificato come indirizzo di rete IPv4, viene generato un errore e il canale non viene avviato. Su piattaforme che supportano l'indirizzamento mappato IPv6, se un nome di connessione (CONNAME) viene specificato come un indirizzo di rete IPv4, l'indirizzo viene mappato a un indirizzo IPv6. Ad esempio, xxx.xxx.xxx.xxx è associato a ::ffff:xxx.xxx.xxx.xxx. L'utilizzo di indirizzi associati potrebbe richiedere traduttori di protocollo. Ove possibile, evitare l'utilizzo di indirizzi associati.
- Se un indirizzo locale (LOCLADDR) viene specificato come indirizzo IP per un canale, viene utilizzato lo stack per tale indirizzo IP. Se l'indirizzo locale (LOCLADDR) viene specificato come un nome host che si risolve in entrambi gli indirizzi IPv4 e IPv6, il nome della connessione (CONNAME) determina quale degli stack viene utilizzato. Se sia l'indirizzo locale (LOCLADDR) che il nome connessione (CONNAME) vengono specificati come nomi host che si risolvono in entrambi gli indirizzi IPv4 e IPv6, lo stack utilizzato viene determinato dall'attributo del gestore code IPADDRV.
- Se il sistema dispone di stack IPv4 e IPv6 doppi configurati e un indirizzo locale (LOCLADDR) non è specificato per un canale, il nome connessione (CONNAME) specificato per il canale determina quale stack IP utilizzare. Se il nome della connessione (CONNAME) viene specificato come un nome host che si risolve in entrambi gli indirizzi IPv4 e IPv6, lo stack utilizzato viene determinato dall'attributo del gestore code IPADDRV.

Mutti Su Multipiattaforme, è possibile impostare un valore di indirizzo locale predefinito che viene utilizzato per tutti i canali mittenti che non hanno un indirizzo locale definito. Il valore predefinito viene definito impostando la variabile di ambiente MQ_LCLADDR prima di avviare il gestore code. Il formato del valore corrisponde a quello dell'attributo MQSC LOCLADDR.

Indirizzi locali con canali mittente cluster

I canali mittenti del cluster ereditano sempre la configurazione del canale ricevente del cluster corrispondente come definito sul gestore code di destinazione. Ciò è vero anche se esiste un canale mittente cluster definito localmente con lo stesso nome, nel qual caso la definizione manuale viene utilizzata solo per la comunicazione iniziale.

Per questo motivo, non è possibile dipendere dal LOCLADDR definito nel canale ricevente del cluster in quanto è probabile che l'indirizzo IP non sia di proprietà del sistema in cui vengono creati i mittenti del cluster. Per questo motivo, LOCLADDR sul ricevente del cluster non deve essere utilizzato a meno che non vi sia un motivo per limitare solo le porte ma non l'indirizzo IP per tutti i potenziali mittenti del cluster ed è noto che tali porte sono disponibili su tutti i sistemi in cui è possibile creare un canale mittente del cluster.

Se un cluster deve utilizzare LOCLADDR per richiamare i canali di comunicazione in uscita da collegare a un indirizzo IP specifico, utilizzare un' Uscita definizione automatica del canale oppure utilizzare il LOCLADDR predefinito per il gestore code, quando possibile. Quando si utilizza un'uscita del canale, forza il valore LOCLADDR dall'uscita in uno qualsiasi dei canali CLUSSDR definiti automaticamente.

Se si utilizza un LOCLADDR non predefinito per i canali mittenti del cluster tramite l'utilizzo di un'uscita o di un valore predefinito, qualsiasi canale mittente del cluster definito manualmente corrispondente, ad esempio a un gestore code del repository completo, deve avere anche il valore LOCLADDR impostato per consentire la comunicazione iniziale sul canale.

Nota: Se il sistema operativo restituisce un errore di bind per la porta fornita in LOCLADDR (o tutte le porte, se viene fornito un intervallo di porte), il canale non viene avviato; il sistema emette un messaggio di errore.

LOCLADDR per canali AMQP

I canali AMQP supportano un formato diverso di LOCLADDR rispetto ad altri canali IBM MQ :

LOCLADDR (*ip-addr*)

LOCLADDR è l'indirizzo di comunicazione locale per il canale. Utilizzare questo parametro se si desidera forzare il client ad utilizzare un particolare indirizzo IP. LOCLADDR è utile anche per forzare un canale ad utilizzare un indirizzo IPv4 o IPv6 se è disponibile una scelta oppure per utilizzare un particolare adattatore di rete su un sistema con più adattatori di rete.

La lunghezza massima di LOCLADDR è MQ_LOCAL_ADDRESS_LENGTH.

Se si omette LOCLADDR, viene automaticamente assegnato un indirizzo locale.

indirizzo ip

ip-addr è un indirizzo di rete singolo, specificato in uno dei tre formati:

decimale puntata IPv4

Ad esempio 192.0.2.1

notazione esadecimale IPv6

Ad esempio 2001:DB8:0:0:0:0:0:0

Formato nome host alfanumerico

Ad esempio WWW.EXAMPLE.COM

Se viene immesso un indirizzo IP, viene convalidato solo il formato dell'indirizzo. L'indirizzo IP non viene convalidato.

Consultare [Gestione dei canali mittenti del cluster definiti automaticamente](#) per ulteriori informazioni.

LONGRTY (Conteggio tentativi lunghi)


Questo attributo specifica il numero massimo di volte in cui il canale tenta di assegnare una sessione al partner.


L'attributo **long retry count** può essere impostato da zero a 999 999 999.

Questo attributo è valido per i seguenti tipi di canale:

- Mittente
- Server
- Mittente del cluster
- Ricevente del cluster

Se il tentativo di assegnazione iniziale ha esito negativo, il numero di *tentativi brevi* viene ridotto e il canale ritenta il numero di volte rimanente. Se ha ancora esito negativo, ritenta un *conteggio tentativi lunghi* numero di volte con un intervallo di *intervallo tentativi lunghi* tra un tentativo e l'altro. Se ha ancora esito negativo, il canale viene chiuso. Il canale deve essere riavviato con un comando; non viene avviato automaticamente dall'iniziatore del canale.

 Su z/OS, un canale non può immettere un nuovo tentativo se è stato superato il numero massimo di canali (**MAXCHL**).

 Su [Multiplatforme](#), per poter ritentare, è necessario che un iniziatore di canali sia in esecuzione. L'iniziatore di canali deve monitorare la coda di iniziazione specificata nella definizione della coda di trasmissione utilizzata dal canale.

Se l'iniziatore del canale (su z/OS) o il canale (su [Multiplatforme](#)) viene arrestato mentre il canale sta eseguendo un nuovo tentativo, il *conteggio tentativi brevi* e il *conteggio tentativi lunghi* vengono

reimpostati quando l'iniziatore del canale o il canale vengono riavviati o quando un messaggio viene correttamente inserito nel canale mittente. Tuttavia, se l'iniziatore del canale (su z/OS) o il gestore code (su Multiplatforms) viene arrestato e riavviato, il *conteggio tentativi brevi* e il *conteggio tentativi lunghi* non vengono reimpostati. Il canale conserva i valori di conteggio dei tentativi che aveva prima del riavvio del gestore code o dell'inserimento del messaggio.

Multi

Su Multiplatforme:

1. Quando un canale passa dallo stato RERETRY allo stato RUNNING, il *conteggio tentativi brevi* e il *conteggio tentativi lunghi* non vengono reimpostati immediatamente. Vengono reimpostati solo quando il primo messaggio passa attraverso il canale con esito positivo dopo che il canale è passato allo stato RUNNING, ossia quando il canale locale conferma il numero di messaggi inviati all'altra estremità.
2. Il *conteggio tentativi brevi* e il *conteggio tentativi lunghi* vengono reimpostati quando il canale viene riavviato.

LONGTMR (Intervallo nuovo tentativo lungo)

Questo attributo è l'intervallo approssimativo in secondi che il canale deve attendere prima di tentare nuovamente di stabilire una connessione, durante la modalità di ripetizione lunga.

L'intervallo tra i tentativi può essere esteso se il canale deve attendere per diventare attivo.

Il canale tenta di connettersi *conteggio tentativi lunghi* numero di volte in questo intervallo lungo, dopo aver tentato il *conteggio tentativi brevi* numero di volte nell'intervallo breve.

Questo attributo può essere impostato da zero a 999 999.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Mittente del cluster
- Ricevente del cluster

Attributi del canale per le parole chiave MQSC (M)

Un elenco alfabetico degli attributi di canale per le parole chiave MQSC, iniziando con la lettera *M*.

MAXINST (Numero massimo di istanze)

Questo attributo specifica il numero massimo di istanze simultanee di un canale di connessione server o di un canale AMQP che è possibile avviare.

Numero massimo di istanze di connessioni del canale di connessione server

Per un canale di connessione server, questo attributo specifica il numero massimo di istanze simultanee di un canale di connessione server che è possibile avviare.

Questo attributo può essere impostato da zero a 999 999 999. Il valore zero indica che su questo canale non è consentita alcuna connessione client. Il valore predefinito è 999 999 999.

Se il valore viene ridotto in modo che sia inferiore al numero di istanze del canale di connessione server attualmente in esecuzione, i canali in esecuzione non vengono influenzati. Tuttavia, le nuove istanze non possono essere avviate fino a quando non cessano di essere eseguite.

Numero massimo di istanze di connessioni canale AMQP

Per un canale AMQP, questo attributo specifica il numero massimo di istanze simultanee di un canale AMQP che possono essere avviate.

Questo attributo può essere impostato da zero a 999 999 999. Il valore zero indica che su questo canale non è consentita alcuna connessione client. Il valore predefinito è 999 999 999.

Se un client tenta di collegarsi e il numero di client connessi ha raggiunto MAXINST, il canale chiude la connessione con un frame di chiusura. Il frame di chiusura contiene il seguente messaggio:

```
amqp:resource-limit-exceeded
```

Se un client si connette con un ID già connesso (ossia esegue un takeover client), il takeover avrà esito positivo indipendentemente dal fatto che il numero di client connessi abbia raggiunto MAXINST.

Per ulteriori informazioni, vedi [Limiti del canale di connessione server](#).

MAXINSTC (numero massimo di istanze per client)

Questo attributo specifica il numero massimo di istanze simultanei di un canale di connessione server che possono essere avviate da un singolo client.

Questo attributo può essere impostato da zero a 999 999 999. Il valore zero indica che su questo canale non è consentita alcuna connessione client. Il valore predefinito è 999 999 999.

Se il valore viene ridotto in modo che sia inferiore al numero di istanze del canale di connessione server attualmente in esecuzione da singoli client, i canali in esecuzione non vengono interessati. Tuttavia, le nuove istanze da tali client non possono essere avviate fino a quando non cessa l'esecuzione di quelle esistenti.

Questo attributo è valido solo per i canali di connessione al server.

Per ulteriori informazioni, vedi [Limiti del canale di connessione server](#).

MAXMSGL (lunghezza messaggio massima)

Questo attributo specifica la lunghezza massima di un messaggio che può essere trasmesso sul canale.

Multi Su Multipiattaforme, specificare un valore maggiore o uguale a zero e minore o uguale alla lunghezza massima del messaggio per il gestore code. Per ulteriori informazioni, consultare il parametro MAXMSGL del comando ALTER QMGR in [ALTER QMGR](#).

z/OS Su IBM MQ for z/OS, specificare un valore maggiore o uguale a zero e minore o uguale a 104 857 600 byte (ossia 100 MB).

Poiché esistono varie implementazioni di sistemi IBM MQ su piattaforme differenti, la dimensione disponibile per l'elaborazione dei messaggi potrebbe essere limitata in alcune applicazioni. Questo numero deve riflettere una dimensione che il proprio sistema può gestire senza stress. Quando un canale viene avviato, viene preso il numero più basso tra i due ad ogni estremità del canale.

Nota: È possibile utilizzare una dimensione massima di messaggi pari a 0 per il canale, che indica che la dimensione deve essere impostata sul valore massimo del gestore code locale.

Aggiungendo la firma digitale e la chiave al messaggio, [Advanced Message Security](#) aumenta la lunghezza del messaggio.

Questo attributo è valido per tutti i tipi di canale.

MCANAME (nome agent canale dei messaggi)

Questo attributo è riservato, se specificato deve essere impostato solo su spazi vuoti e ha una lunghezza massima di 20 caratteri.

MCATYPE (Tipo agent canale messaggi)

Questo attributo può specificare l'agent del canale dei messaggi come un processo o un thread.

I vantaggi dell'esecuzione come processo includono:

- Isolamento per ogni canale che fornisce una maggiore integrità
- Autorizzazione lavoro specifica per ogni canale
- Controllo sulla pianificazione del lavoro

I vantaggi dei thread includono:

- Utilizzo molto ridotto dello storage
- Configurazione più semplice digitando sulla riga comandi
- Esecuzione più veloce - è più rapido avviare un thread che istruire il sistema operativo per avviare un processo

Nota: Per i tipi di canale mittente, server e richiedente, il valore predefinito è `processo`. Per i tipi di canale mittente e destinatario del cluster, il valore predefinito è `thread`. Questi valori predefiniti possono cambiare durante l'installazione.

Se si specifica `process` nella definizione del canale, viene avviato un processo `RUNMQCHL`. Se si specifica `thread`, l'MCA viene eseguito su un thread del processo `AMQRMPPA` o del processo `RUNMQCHI` se viene specificato `MQNOREMPOOL`. Sulla macchina che riceve le allocazioni in entrata, l'MCA viene eseguito come un thread se si utilizza `RUNMQLSR`. Viene eseguito come un processo se si utilizza `inetd`.

z/OS Su IBM MQ for z/OS, questo attributo viene supportato solo per canali con un tipo di canale ricevente del cluster.

Multi Su altre piattaforme, questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Richiedente
- Mittente del cluster
- Ricevente del cluster

MCAUSER (Identificativo utente agent canale messaggi)

Questo attributo rappresenta l'identificativo utente (una stringa) che deve essere utilizzato da MCA per l'autorizzazione ad accedere alle risorse IBM MQ.

Nota: Un modo alternativo per fornire un ID utente per un canale da eseguire consiste nell'utilizzare i record di autenticazione di canale. Con i record di autenticazione di canale, connessioni differenti possono utilizzare lo stesso canale utilizzando credenziali differenti. Se `MCAUSER` sul canale è impostato e i record di autenticazione di canale vengono utilizzati per essere applicati allo stesso canale, i record di autenticazione di canale hanno la precedenza. Il `MCAUSER` sulla definizione di canale viene utilizzato solo se il record di autenticazione di canale utilizza `USERSRC (CHANNEL)`.

Questa autorizzazione include (se l'autorizzazione `PUT` è `DEF`) l'inserimento del messaggio nella coda di destinazione per i canali riceventi o richiedenti.

Su IBM MQ for Windows, l'identificativo utente può essere qualificato dal dominio utilizzando il formato, `user@domain`, dove `domain` deve essere il dominio dei sistemi Windows del sistema locale o un dominio attendibile.

Se questo attributo è vuoto, l'MCA utilizza il proprio identificativo utente predefinito. Per ulteriori informazioni, vedere `DEFINE CHANNEL`.

Questo attributo è valido per i tipi di canale:

- Ricevente
- Richiedente
- Connessione server

- Ricevente del cluster

MODENAME (nome modalità LU 6.2)

Questo attributo è per l'utilizzo con connessioni LU 6.2. Dà una definizione extra per le caratteristiche di sessione della connessione quando viene eseguita un'assegnazione di sessione di comunicazione.

Quando si utilizzano le informazioni laterali per le comunicazioni SNA, il nome del modo è definito nell'oggetto lato comunicazioni CPI-C o nelle informazioni lato APPC e questo attributo deve essere lasciato vuoto; altrimenti, deve essere impostato sul nome del modo SNA.

Il nome deve essere lungo da uno a otto caratteri alfanumerici.

Questo attributo è valido solo per i tipi di canale:

- Mittente
- Server
- Richiedente
- Connessione client
- Mittente del cluster
- Ricevente del cluster

MONCHL (Monitoraggio)

Questo attributo controlla la raccolta dei dati di monitoraggio in linea.

I possibili valori sono:

QMGR

La raccolta di Online Monitoring Data viene ereditata dall'impostazione dell'attributo MONCHL nell'oggetto gestore code. Questo è il valore predefinito.

Disattivo

La raccolta di dati di monitoraggio online per questo canale è disabilitata.

BASSO

Un basso rapporto di raccolta dati con un effetto minimo sulle prestazioni. Tuttavia, i risultati di monitoraggio visualizzati potrebbero non essere aggiornati.

MEDIO

Un rapporto moderato di raccolta dati con un effetto limitato sulle prestazioni del sistema.

ALTO

Un alto rapporto di raccolta dati con la possibilità di un effetto sulle prestazioni. Tuttavia, i risultati di monitoraggio mostrati sono i più attuali.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Ricevente
- Richiedente
- Connessione server
- Mittente del cluster
- Ricevente del cluster

Per ulteriori informazioni sul monitoraggio dei dati, consultare [Visualizzazione dei dati di monitoraggio della coda e del canale](#).

MRDATA (Dati utente uscita nuovo tentativo messaggio)

il suo attributo specifica i dati passati all'exit dei tentativi dei messaggi del canale quando viene richiamato.

Questo attributo è valido per i tipi di canale:

- Ricevente
- Richiedente
- Ricevente del cluster

MREXIT (Nome uscita nuovo tentativo messaggio)

Questo attributo specifica il nome del programma di uscita utente che deve essere eseguito dall'uscita utente di ripetizione del messaggio.

Lasciare vuoto se non è attivo alcun programma di uscita per nuovi tentativi di messaggi.

Il formato e la lunghezza massima del nome dipendono dalla piattaforma, come per [“RCVEXIT \(Nome uscita ricezione\)”](#) a pagina 120. Tuttavia, è possibile specificare solo un'uscita di nuovo tentativo messaggio.

Questo attributo è valido per i tipi di canale:

- Ricevente
- Richiedente
- Ricevente del cluster

MRRTY (Conteggio tentativi messaggi)

Questo attributo specifica il numero di volte in cui il canale tenta di ridistribuire il messaggio.

Questo attributo controlla l'azione dell'MCA solo se il nome dell'uscita del nuovo tentativo di messaggio è vuoto. Se il nome dell'uscita non è vuoto, il valore di MRRTY viene passato all'uscita, ma il numero di tentativi effettuati (se presenti) è controllato dall'uscita e non da questo attributo.

Il valore deve essere compreso tra 0 e 999 999 999. Il valore zero indica che non vengono effettuati ulteriori tentativi. Il valore predefinito è 10.

Questo attributo è valido per i tipi di canale:

- Ricevente
- Richiedente
- Ricevente del cluster

MRTMR (Intervallo tentativi messaggi)

Questo attributo specifica l'intervallo minimo di tempo, in millisecondi, che deve trascorrere prima che il canale possa ritentare l'operazione MQPUT.

Questo attributo controlla l'azione dell'MCA solo se il nome dell'uscita del nuovo tentativo di messaggio è vuoto. Se il nome dell'uscita non è vuoto, il valore di MRTMR viene inoltrato all'uscita per essere utilizzato dall'uscita, ma l'intervallo dei tentativi viene controllato dall'uscita e non da questo attributo.

Il valore deve essere compreso tra 0 e 999 999 999. Un valore zero indica che il nuovo tentativo viene eseguito appena possibile (se il valore di MRRTY è maggiore di zero). Il valore predefinito è 1000.

Questo attributo è valido per i seguenti tipi di canale:

- Ricevente
- Richiedente
- Ricevente del cluster

MSGDATA (Dati utente uscita messaggio)

Questo attributo specifica i dati utente passati alle uscite dei messaggi del canale.

È possibile eseguire una sequenza di uscite di messaggi. Le limitazioni sulla lunghezza dei dati utente e un esempio di come specificare MSGDATA per più di un'uscita sono quelle mostrate per RCVDATA. Consultare [“RCVDATA \(Dati utente uscita ricezione\)”](#) a pagina 119.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Ricevente
- Richiedente
- Mittente del cluster
- Ricevente del cluster

MSGEXIT (Nome uscita messaggio)

Questo attributo indica il nome del programma di uscita utente che deve essere eseguito dall'uscita messaggi del canale.

Questo attributo può essere un elenco di nomi di programmi da eseguire in successione. Lasciare vuoto se non è attiva alcuna uscita del messaggio del canale.

Il formato e la lunghezza massima di questo attributo dipendono dalla piattaforma, come per [“RCVEXIT \(Nome uscita ricezione\)”](#) a pagina 120.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Ricevente
- Richiedente
- Mittente del cluster
- Ricevente del cluster

Attributi canale per le parole chiave MQSC (N-R)

Un elenco in ordine alfabetico degli attributi di canale per le parole chiave MQSC, iniziando con le lettere da *N* a *R*.

NETPRTY (Priorità connessione di rete)

L'attributo del canale NETPRTY specifica la priorità per un canale CLUSRCVR . Il valore deve essere compreso nell'intervallo 0-9, dove 0 è la priorità più bassa e 9 è la più alta.

Utilizzare l'attributo NETPRTY per rendere una rete la rete principale e un'altra la rete di backup. Dato un insieme di canali ugualmente classificati, il clustering sceglie il percorso con la priorità più alta quando sono disponibili più percorsi.

Un esempio tipico di utilizzo dell'attributo del canale NETPRTY è quello di differenziare le reti che hanno costi o velocità differenti e conettono le stesse destinazioni.

Nota: Specificare questo attributo sul canale ricevente del cluster sul gestore code di destinazione. È probabile che qualsiasi bilanciamento specificato sul canale mittente del cluster corrispondente venga ignorato. Consultare [Canali cluster](#).

NPMSPEED (Velocità messaggi non persistenti)

Questo attributo specifica la velocità con cui devono essere inviati i messaggi non persistenti.

I possibili valori sono:

NORMALE


I messaggi non persistenti su un canale vengono trasferiti all'interno delle transazioni.

VELOCE

I messaggi non persistenti su un canale non vengono trasferiti all'interno delle transazioni.

Il valore predefinito è FAST. Il vantaggio è che i messaggi non persistenti diventano disponibili per il richiamo molto più rapidamente. Lo svantaggio è che poiché non fanno parte di una transazione, i messaggi potrebbero andare persi se si verifica un errore di trasmissione o se il canale si arresta quando i messaggi sono in transito. Consultare [Sicurezza dei messaggi](#).

Note:

1.  Se i log di ripristino attivi per IBM MQ for z/OS stanno effettuando la commutazione e l'archiviazione più frequentemente del previsto, dato che i messaggi inviati attraverso un canale sono non persistenti, l'impostazione di NPMSPEED (FAST) sia sull'estremità di invio che su quella di ricezione del canale può ridurre al minimo il SISTEMA SYSTEM.CHANNEL.SYNCQ aggiorna.
2. Se si sta riscontrando un utilizzo elevato della CPU relativo agli aggiornamenti al SISTEMA SYSTEM.CHANNEL.SYNCQ, l'impostazione di NPMSPEED (FAST) può ridurre in modo significativo l'uso della CPU.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Ricevente
- Richiedente
- Mittente del cluster
- Ricevente del cluster

Password (Password)

Questo attributo specifica una password che può essere utilizzata da MCA quando si tenta di avviare una sessione LU sicura 6.2 con un MCA remoto.

È possibile specificare una password di lunghezza massima di 12 caratteri, anche se vengono utilizzati solo i primi 10 caratteri.

 Su IBM MQ for z/OS, questo attributo è valido solo per canali di connessione client.

 Su altre piattaforme, questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Richiedente
- Connessione client
- Mittente del cluster

PORT (Numero porta)

Specificare il numero di porta utilizzato per collegare il client AMQP.

La porta predefinita per le connessioni AMQP 1.0 è 5672. Se si sta già utilizzando la porta 5672, è possibile specificare una porta differente.

PUTAUT (autorizzazione PUT)

Questo attributo specifica il tipo di elaborazione di sicurezza che deve essere eseguita da MCA.

Utilizzare questo attributo per scegliere il tipo di elaborazione di sicurezza che deve essere eseguita da MCA durante l'esecuzione:

- Un comando MQPUT per la coda di destinazione (per i canali di messaggi) o
- Una chiamata MQI (per i canali MQI).

z/OS Su z/OS, gli ID utente controllati e il numero di ID utente controllati dipendono dall'impostazione del profilo della RACF classe hlq.RESLEVEL MQADMIN. A seconda del livello di accesso l'ID utente dell'iniziatore di canali deve essere hlq.RESLEVEL, zero, vengono controllati uno o due ID utente. Per vedere quanti ID utente vengono controllati, consultare [RESLEVEL](#) e le [connessioni dell'iniziatore di canali](#). Per ulteriori informazioni su quali ID utente vengono controllati, consultare [ID utente utilizzati dall'iniziatore di canali](#).

È possibile scegliere tra le opzioni seguenti:

Sicurezza del processo, denominata anche autorizzazione predefinita (DEF)

Viene utilizzato l'ID utente predefinito.

Multi Su Multiplatforms, l'ID utente utilizzato per controllare l'autorizzazione aperta sulla coda è quello del processo o dell'utente che esegue l'MCA all'estremità di ricezione del canale dei messaggi.

z/OS Su z/OS, è possibile utilizzare l'ID utente ricevuto dalla rete e l'ID utente derivato da [MCAUSER](#), in base al numero di ID utente da controllare.

Le code vengono aperte con l'ID utente e l'opzione di apertura MQOO_SET_ALL_CONTEXT.

CTX (Context Security)

L'ID utente dalle informazioni di contesto associate al messaggio viene utilizzato come ID utente alternativo.

Il *UserIdentifier* nella descrizione del messaggio viene spostato nel campo *AlternateUserId* nel descrittore dell'oggetto. La coda viene aperta con le opzioni di apertura MQOO_SET_ALL_CONTEXT e MQOO_ALTERNATE_USER_AUTHORITY.

Multi Su Multiplatforms, l'ID utente utilizzato per controllare l'autorizzazione di apertura sulla coda per MQOO_SET_ALL_CONTEXT e MQOO_ALTERNATE_USER_AUTHORITY è quello del processo o dell'utente che esegue l'MCA all'estremità ricevente del canale dei messaggi. L'ID utente utilizzato per controllare l'autorizzazione aperta sulla coda per MQOO_OUTPUT è *UserIdentifier* nel descrittore del messaggio.

z/OS Su z/OS, l'ID utente ricevuto dalla rete o quello derivato da [MCAUSER](#) potrebbe essere utilizzato, così come l'ID utente dalle informazioni di contesto nel descrittore del messaggio, in base al numero di ID utente da controllare.

CTX (Context Security) non supportato sui canali di connessione server.

z/OS Solo sicurezza agente canale dei messaggi (ONLYMCA)

Viene utilizzato l'ID utente derivato da [MCAUSER](#).

Le code vengono aperte con l'opzione di apertura MQOO_SET_ALL_CONTEXT.

Questo valore si applica solo a z/OS.




z/OS ALTMCA (Alternare Message Channel Agent security)

L'ID utente dalle informazioni di contesto (il campo *UserIdentifier*) nel descrittore del messaggio potrebbe essere utilizzato, così come l'ID utente derivato da [MCAUSER](#), in base al numero di ID utente che devono essere controllati.


Questo valore si applica solo a z/OS.

Ulteriori dettagli sui campi di contesto e sulle opzioni di apertura sono disponibili in [Controllo delle informazioni di contesto](#).

Ulteriori informazioni sulla sicurezza sono disponibili qui:

- [Protezione](#)
-  [Impostazione della sicurezza su AIX, Linux, and Windows](#)
-  [Impostazione della sicurezza su IBM i](#)
-  [Impostazione della sicurezza su z/OS](#)

Questo attributo è valido per i tipi di canale:

- Ricevente
- Richiedente
-  Connessione server (solo z/OS)
- Ricevente del cluster

QMNAME (Nome gestore code)

il suo attributo specifica il nome del gestore code o del gruppo di gestori code a cui un'applicazione IBM MQ MQI client può richiedere la connessione.

Questo attributo è valido per i tipi di canale:

- Connessione client

QSGDISP (Disposizione)



Questo attributo specifica la disposizione del canale in un gruppo di condivisione code. È valido solo su z/OS .

I valori sono:

QMGR

Il canale è definito nella serie di pagine del gestore code che esegue il comando. Questo è il valore predefinito.

GRUPPO

Il canale è definito nel repository condiviso. Questo valore è consentito solo se è presente un ambiente del gestore code condiviso. Quando un canale viene definito con QSGDISP (GROUP), il comando DEFINE CHANNEL (name) NOREPLACE QSGDISP (COPY) viene generato automaticamente e inviato a tutti i gestori code attivi per fare in modo che creino copie locali sulla serie di pagine 0. Per i gestori code che non sono attivi o che si uniscono al gruppo di condivisione code in un secondo momento, il comando viene generato all'avvio del gestore code.

Copia

Il canale è definito nella serie di pagine del gestore code che esegue il comando, copiando le relative definizioni dal canale QSGDISP (GROUP) con lo stesso nome. Questo valore è consentito solo se è presente un ambiente del gestore code condiviso.

Questo attributo è valido per tutti i tipi di canale.

RCVDATA (Dati utente uscita ricezione)

Questo attributo specifica i dati utente passati all'uscita di ricezione.

È possibile eseguire una sequenza di uscite di ricezione. La stringa di dati utente per una serie di uscite deve essere separata da una virgola, da spazi o da entrambi. Ad esempio:

```
RCVDATA(exit1_data exit2_data)
MSGDATA(exit1_data,exit2_data)
SENDDATA(exit1_data, exit2_data)
```

ALW Nei sistemi IBM MQ for UNIX e Windows , la lunghezza della stringa dei nomi di uscita e delle stringhe dei dati utente è limitata a 500 caratteri.

IBM i In IBM MQ for IBM i, è possibile specificare fino a 10 nomi di uscita e la lunghezza dei dati utente per ciascuno è limitata a 32 caratteri.

z/OS In IBM MQ for z/OS, è possibile specificare fino a otto stringhe di dati utente, ognuna con una lunghezza di 32 caratteri.

Questo attributo è valido per tutti i tipi di canale.

RCVEXIT (Nome uscita ricezione)

Questo attributo specifica il nome del programma di uscita utente che deve essere eseguito dall'uscita utente di ricezione del canale.

Questo attributo può essere un elenco di nomi di programmi da eseguire in successione. Lasciare vuoto se non è attiva alcuna uscita utente di ricezione del canale.

Il formato e la lunghezza massima di questo attributo dipendono dalla piattaforma:

- z/OS** Su z/OS è un nome modulo di caricamento, con una lunghezza massima di 8 caratteri, ad eccezione dei canali di connessione client in cui la lunghezza massima è 128 caratteri.
- IBM i** Su IBM i, è nel formato:

```
libname/progname
```

quando specificato nei comandi CL.

Quando specificato in MQSC (IBM MQ Commands), ha il formato:

```
progname libname
```

dove *progname* occupa i primi 10 caratteri e *libname* i secondi 10 caratteri (entrambi riempiti di spazi a destra, se necessario). La lunghezza massima della stringa è 20 caratteri.

- Linux** **AIX** Su AIX and Linux, è nel formato:

```
libraryname(functionname)
```

La lunghezza massima della stringa è 40 caratteri.

- Windows** Su Windows, è nel formato:

```
dllname(functionname)
```

dove *dllname* è specificato senza il suffisso .DLL. La lunghezza massima della stringa è 40 caratteri.

- z/OS** Durante la definizione automatica del canale mittente del cluster su z/OS, i nomi delle uscite del canale vengono convertiti in formato z/OS . Se si desidera controllare la modalità di conversione dei nomi di uscita, è possibile scrivere un'uscita di definizione automatica del canale. Per ulteriori informazioni, consultare [Programma di uscita di definizione automatica del canale](#).

È possibile specificare un elenco di nomi di programmi di uscita messaggi, di invio o di ricezione. I nomi devono essere separati da una virgola, uno spazio o entrambi. Ad esempio:

```
RCVEXIT(exit1 exit2)
MSGEXIT(exit1,exit2)
SENDEXIT(exit1, exit2)
```

La lunghezza totale della stringa dei nomi di uscita e delle stringhe dei dati utente per un particolare tipo di uscita è limitata a 500 caratteri.

- **IBM i** Su IBM MQ for IBM i, è possibile elencare fino a 10 nomi di uscita.
- **z/OS** In IBM MQ for z/OS, è possibile elencare fino a otto nomi di uscita.

Questo attributo è valido per tutti i tipi di canale.

Attributi canale per le parole chiave MQSC (S)

Un elenco alfabetico degli attributi di canale per le parole chiave MQSC, iniziando con la lettera S.

SCYDATA (Dati utente uscita di sicurezza)

Questo attributo specifica i dati utente passati all'uscita di sicurezza.

La lunghezza massima è 32 caratteri.

Questo attributo è valido per tutti i tipi di canale.

SCYEXIT (Nome uscita di sicurezza)

Questo attributo specifica il nome del programma di uscita che deve essere eseguito dall'uscita di sicurezza del canale.

Lasciare vuoto se non è attiva alcuna uscita di sicurezza del canale.

Il formato e la lunghezza massima del nome dipendono dalla piattaforma, come per [RCVEXIT](#). Tuttavia, è possibile specificare una sola uscita di sicurezza.

Per ulteriori informazioni sulle uscite di sicurezza, consultare [Programmi di uscita di sicurezza del canale](#).

Questo attributo è valido per tutti i tipi di canale.

SENDDATA (Invio dati utente uscita)

Questo attributo specifica i dati utente passati all'uscita di invio.

È possibile eseguire una sequenza di uscite di invio. Le limitazioni sulla lunghezza dei dati utente e un esempio di come specificare SENDDATA per più di un'uscita, sono come mostrato per RCVDATA. Vedere [RCVDATA](#).

Questo attributo è valido per tutti i tipi di canale.

SENDEXIT (Nome uscita invio)

Questo attributo specifica il nome del programma di uscita che deve essere eseguito dall'uscita di trasmissione del canale.

Questo attributo può essere un elenco di nomi di programmi che devono essere eseguiti in sequenza. Lasciare vuoto se non è attiva alcuna uscita di invio del canale.

Il formato e la lunghezza massima di questo attributo dipendono dalla piattaforma, come per [RCVEXIT](#).

Questo attributo è valido per tutti i tipi di canale.

SEQWRAP (ritorno a capo numero di sequenza)

Questo attributo specifica il numero più alto che il numero di sequenza del messaggio raggiunge prima di riavviarlo a 1.

Il valore del numero deve essere abbastanza alto da evitare che un numero venga riemesso mentre viene ancora utilizzato da un messaggio precedente. Le due estremità di un canale devono avere lo stesso valore di ritorno a capo del numero di sequenza quando si avvia un canale; altrimenti, si verifica un errore.

Il valore può essere impostato da 100 a 999 999 999.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Ricevente
- Richiedente
- Mittente del cluster
- Ricevente del cluster

SHORTRTY (conteggio tentativi brevi)

Questo attributo specifica il numero massimo di volte in cui il canale tenta di assegnare una sessione al partner.

L'attributo SHORTRTY può essere impostato da zero a 999 999 999.

Se sono stati definiti più indirizzi IP all'interno del canale e la riconnessione è necessaria, IBM MQ valuta la definizione del canale e tenta di connettersi a ciascun indirizzo IP nell'ordine in cui è stato definito fino a quando non viene stabilita una connessione riuscita o fino a quando non sono stati tentati tutti gli indirizzi.

In questo caso, SHORTRTY si riferisce al numero di tentativi totali che il canale globale tenta di riconnettere e non ai singoli indirizzi IP

Se il tentativo di assegnazione iniziale non riesce, il *conteggio tentativi brevi* viene ridotto e il canale ritenta il numero rimanente di volte con un intervallo, definito nell'attributo **short retry interval**, tra un tentativo e l'altro. Se ha ancora esito negativo, ritenta *conteggio tentativi lunghi* il numero di volte con un intervallo di *intervallo tentativi lunghi* tra un tentativo e l'altro. Se ha ancora esito negativo, il canale viene chiuso.

z/OS Su z/OS, un canale non può immettere un nuovo tentativo se è stato superato il numero massimo di canali (**MAXCHL**).

Multi Su Multipiattaforme, per poter ritentare, è necessario che un iniziatore di canali sia in esecuzione. L'iniziatore di canali deve monitorare la coda di iniziazione specificata nella definizione della coda di trasmissione utilizzata dal canale.

Se l'iniziatore di canali (su z/OS) o il canale (su Multipiattaforme) viene arrestato mentre il canale viene riprovato, il *conteggio tentativi brevi* e il *conteggio tentativi lunghi* vengono reimpostati quando l'iniziatore del canale o il canale vengono riavviati o quando un messaggio viene correttamente inserito nel canale mittente. Tuttavia, se l'iniziatore del canale (su z/OS) o il gestore code (su Multipiattaforme) viene arrestato e riavviato, il *conteggio tentativi brevi* e il *conteggio tentativi lunghi* non vengono reimpostati. Il canale conserva i valori di conteggio dei tentativi che aveva prima del riavvio del gestore code o dell'inserimento del messaggio.

Multi Su Multipiattaforme:

1. Quando un canale passa dallo stato RERETRY allo stato RUNNING, il *conteggio tentativi brevi* e il *conteggio tentativi lunghi* non vengono reimpostati immediatamente. Vengono reimpostati solo quando

il primo messaggio passa attraverso il canale con esito positivo dopo che il canale è passato allo stato RUNNING, ossia quando il canale locale conferma il numero di messaggi inviati all'altra estremità.

2. Il *conteggio tentativi brevi* e il *conteggio tentativi lunghi* vengono reimpostati quando il canale viene riavviato.

Questo attributo è valido per i seguenti tipi di canale:

- Mittente
- Server
- Mittente del cluster
- Ricevente del cluster

SHORTTMR (Intervallo nuovo tentativo breve)

Questo attributo specifica l'intervallo approssimativo, in secondi, che il canale deve attendere prima di tentare nuovamente di stabilire la connessione, durante la modalità di nuovo tentativo breve.

L'intervallo tra i tentativi potrebbe essere esteso se il canale deve attendere per diventare attivo.

Questo attributo può essere impostato da zero a 999 999.

Se sono stati definiti più indirizzi IP all'interno del canale e la riconnessione è necessaria, IBM MQ valuta la definizione del canale e tenta di connettersi a ciascun indirizzo IP nell'ordine in cui è stato definito fino a quando non viene stabilita una connessione riuscita o fino a quando non sono stati tentati tutti gli indirizzi.

In questo caso, SHORTTMR si riferisce al tempo di attesa del canale complessivo per riavviare il processo di connessione e non ai singoli indirizzi IP.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Mittente del cluster
- Ricevente del cluster

SPLPROT (Protezione della politica di sicurezza)



Questo attributo specifica il modo in cui un MCA (Message Channel Agent) da server a server deve gestire la protezione dei messaggi quando AMS è attivo ed esiste una politica applicabile.

Questo attributo può essere impostato su:

PASSTHRU

Sui canali mittente, server, ricevente e richiedente

ELIMINA

Sui canali mittente e server

ASPOLICY

Sui canali riceventi e richiedenti

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Ricevente
- Richiedente

SSLCAUTH (autenticazione client SSL)

L'attributo **SSLCAUTH** specifica se il canale deve ricevere e autenticare un certificato TLS da un client TLS.

SSLCAUTH è un attributo facoltativo. I valori possibili di questo attributo sono:

Facoltativo

Se il client TLS del peer invia un certificato, il certificato viene elaborato come normale, ma l'autenticazione non ha esito negativo se non viene inviato alcun certificato.

OBBLIGATORIO

Se il client TLS non invia un certificato, l'autenticazione non riesce.

Il valore predefinito è REQUIRED.

È possibile specificare un valore per **SSLCAUTH** in una definizione di canale non TLS. Vale a dire, una definizione di canale su cui l'attributo **SSLCIPH** manca o è vuoto.

Per ulteriori informazioni su **SSLCAUTH**, consultare [DEFINE CHANNEL \(MQTT\)](#) e [Protezione](#).

L'attributo **SSLCAUTH** è valido su tutti i tipi di canale che possono mai ricevere un flusso di avvio del canale, ad eccezione dei canali mittente. Questo attributo è valido per i tipi di canale:

- Server
- Ricevente
- Richiedente
- Connessione server
- Ricevente del cluster

SSLCIPH (specifica di cifratura SSL)


L'attributo **SSLCIPH** specifica un alias o un singolo CipherSpec denominato per una connessione TLS.

Ogni IBM MQ definizione di canale include l'attributo **SSLCIPH**. Il valore è una stringa con una lunghezza massima di 32 caratteri.

L'attributo **SSLCIPH** è valido solo per i canali con un tipo di trasporto (**TRPTYPE**) TCP. Se **TRPTYPE** non è TCP, i dati vengono ignorati e non viene emesso alcun messaggio di errore.

Note:

- L'attributo **SSLCIPH** può contenere un valore vuoto, il che significa che non stai utilizzando TLS. Se un'estremità del canale ha un attributo **SSLCIPH** vuoto, anche l'altra estremità del canale deve avere un attributo **SSLCIPH** vuoto.

 Se è abilitato [SecureCommsOnly](#), la comunicazione di testo semplice non è supportata e l'avvio del canale non riesce.

- In alternativa, se **SSLCIPH** contiene un valore non vuoto, il valore può essere un alias o un CipherSpec denominato. I canali negoziano la CipherSpec più forte supportata da entrambe le estremità del canale.
- Un client .NET completamente gestito può specificare il valore speciale *NEGOZIARE. Questa opzione consente al canale di selezionare la versione di protocollo più recente supportata dal framework .NET e di negoziare una CipherSpec supportata dal server.

L'attributo **SSLCIPH** è valido solo per i canali con un tipo di trasporto (**TRPTYPE**) TCP. Se **TRPTYPE** non è TCP, i dati vengono ignorati e non viene emesso alcun messaggio di errore.

Per ulteriori informazioni su **SSLCIPH**, consultare [DEFINE CHANNEL](#) e [Specifica di CipherSpecs](#).

SSLPEER (peer SSL)

L'attributo **SSLPEER** viene utilizzato per controllare il DN (Distinguished Name) del certificato dal gestore code peer o dal client all'altra estremità di un canale IBM MQ.

Nota: Un modo alternativo per limitare le connessioni nei canali mediante la corrispondenza con il DN (Distinguished Name) del soggetto TLS consiste nell'utilizzare i record di autenticazione di canale. Con i record di autenticazione di canale, è possibile applicare diversi modelli di DN (Distinguished Name) del soggetto TLS allo stesso canale. Se sia **SSLPEER** sul canale che un record di autenticazione di canale vengono utilizzati per essere applicati allo stesso canale, il certificato in entrata deve corrispondere a entrambi i modelli per potersi connettere.

Se il DN ricevuto dal peer non corrisponde al valore **SSLPEER**, il canale non viene avviato.

SSLPEER è un attributo facoltativo. Se non viene specificato un valore, il DN peer non viene controllato quando il canale viene avviato.

La lunghezza massima dell'attributo **SSLPEER** dipende dalla piattaforma:

- ▶ **z/OS** Su z/OS, la lunghezza massima dell'attributo è 256 byte.
- ▶ **Multi** Su tutte le altre piattaforme, è di 1024 byte.

I record di autenticazione di canale forniscono maggiore flessibilità quando si utilizza **SSLPEER** e supportano una lunghezza massima di 1024 byte su tutte le piattaforme.

Il controllo dei valori degli attributi **SSLPEER** dipende anche dalla piattaforma:

- ▶ **z/OS** Su z/OS, i valori di attributo utilizzati non vengono controllati. Se si immettono valori non corretti, il canale non riesce all'avvio e i messaggi di errore vengono scritti nel log degli errori ad entrambe le estremità del canale. Un evento di errore SSL del canale viene generato anche su entrambe le estremità del canale.
- ▶ **Multi** Su Multiplatforme tale supporto **SSLPEER**, la validità della stringa viene controllata quando viene immessa per la prima volta.

È possibile specificare un valore per **SSLPEER** in una definizione di canale non TLS, uno in cui l'attributo **SSLCIPH** è mancante o vuoto. È possibile utilizzare questa opzione per disabilitare temporaneamente TLS per il debug senza dover cancellare e successivamente reimmettere i parametri TLS.

L'attributo **SSLPEER** è valido per tutti i tipi di canale.

Per ulteriori informazioni sull'utilizzo di **SSLPEER**, consultare [SET CHLAUTH, Securing Record di autenticazione di canale](#).

STATCHL (statistiche canale)

Questo attributo controlla la raccolta dei dati statistici per canali.

I valori possibili sono:

QMGR

La raccolta dei dati statistici per questo canale si basa sull'impostazione dell'attributo del gestore code STATCHL. Questo è il valore predefinito.

Disattivo

La raccolta dei dati statistici per questo canale è disabilitata.

BASSO

La raccolta dati statistici per questo canale è abilitata con un rapporto basso di raccolta dati.

MEDIO

La raccolta dati statistici per questo canale è abilitata con un rapporto moderato di raccolta dati.

ALTO

La raccolta dati statistici per questo canale è abilitata con un rapporto elevato di raccolta dati.

Per ulteriori informazioni sulle statistiche del canale, consultare [Riferimento monitoraggio](#).

- ▶ **z/OS** Su z/OS sistemi, abilitando questo parametro si attiva semplicemente la raccolta dei dati statistici, indipendentemente dal valore selezionato. La specifica di LOW, MEDIUM o HIGH non comporta

alcuna differenza sui risultati. Questo parametro deve essere abilitato al fine di raccogliere i record di contabilità di canale.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Ricevente
- Richiedente
- Mittente del cluster
- Ricevente del cluster

Attributi del canale per le parole chiave MQSC (T-Z)

Un elenco in ordine alfabetico degli attributi di canale per le parole chiave MQSC, che inizia con le lettere da T a Z.

TPNAME (nome programma transazione LU 6.2)

Questo attributo è per l'utilizzo con connessioni LU 6.2. È il nome, o il nome generico, del programma di transazione (MCA) da eseguire all'estremità del collegamento.

Quando si utilizzano le informazioni collaterali per le comunicazioni SNA, il nome del programma di transazione viene definito nell'oggetto lato comunicazioni CPI-C o nelle informazioni lato APPC e questo attributo deve essere lasciato vuoto. In caso contrario, questo nome è richiesto dai canali mittenti e richiedenti.

Il nome può essere lungo fino a 64 caratteri.

Il nome deve essere impostato sul nome del programma di transazione SNA, a meno che CONNAME non contenga un nome oggetto laterale, nel qual caso deve essere impostato su spazi vuoti. Il nome effettivo viene preso invece dall'oggetto lato comunicazioni CPI-C o dal dataset di informazioni lato APPC.

Queste informazioni sono impostate in modi diversi su diverse piattaforme; consultare [Configurazione dell'accodamento distribuito](#) per ulteriori informazioni sull'impostazione della comunicazione per la propria piattaforma.

Questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Richiedente
- Connessione client
- Mittente del cluster
- Ricevente del cluster

TPROOT (root argomento)

Questo attributo specifica la root dell'argomento per un canale AMQP.

È possibile utilizzare l'attributo TPROOT per specificare una root argomento per un canale AMQP. L'utilizzo di questo attributo garantisce che un'applicazione MQ Light, quando distribuita a un gestore code, non pubblici o sottoscriva i messaggi verso o dalle aree della struttura ad albero degli argomenti utilizzate da altre applicazioni.

Il valore predefinito per TPROOT è SYSTEM.BASE.TOPIC. Con questo valore, la stringa di argomenti che un client AMQP utilizza per pubblicare o sottoscrivere non ha prefisso e il client può scambiare messaggi con altre applicazioni di pubblicazione / sottoscrizione MQ. Per fare in modo che i client AMQP pubblichino e sottoscrivano sotto un prefisso di argomento, creare prima un oggetto argomento MQ con una stringa di

argomento impostata sul prefisso desiderato, quindi modificare il valore dell'attributo TPROOT del canale AMQP nel nome dell'oggetto argomento MQ creato. Il seguente esempio mostra la root dell'argomento impostata su APPGROUP1.BASE.TOPIC per canale AMQP MYAMQP:

```
DEFINE CHANNEL(MYAMQP) CHLTYPE(AMQP) TPROOT(APPGROUP1.BASE.TOPIC) PORT(5673)
```

Nota: Se il valore dell'attributo TPROOT, o la stringa dell'argomento che lo sottende, viene modificato, gli argomenti AMQP esistenti e i relativi messaggi potrebbero essere orfani.

TRPTYPE (Tipo di trasporto)

Questo attributo specifica il tipo di trasporto da utilizzare.

I valori possibili sono:

Valore	Tipo trasporto
LU62	LU 6.2
TCP	TCP/IP
NETBIOS	NetBIOS "1" a pagina 127
SPX	SPX "1" a pagina 127

Note:

1. Per l'utilizzo su Windows. Può essere utilizzato anche su z/OS per definire canali di collegamento client da utilizzare su Windows.

Questo attributo è valido per tutti i tipi di canale, ma viene ignorato dagli agent del canale messaggi di risposta.

USECLTID (Utilizza ID client)

Specificare se l'ID client viene utilizzato per la connessione su un canale AMQP. Impostare su Yes o No.

USEDLQ (Utilizza coda messaggi non instradabili)

Questo attributo determina se la coda di messaggi non recapitabili (o la coda di messaggi non recapitabili) viene utilizzata quando i messaggi non possono essere consegnati dai canali.

I possibili valori sono:

No

I messaggi che non possono essere consegnati da un canale vengono considerati un errore. Il canale elimina questi messaggi o il canale termina, in conformità con l'impostazione di NPMSPEED.

Sì (impostazione predefinita)

Se l'attributo DEADQ del gestore code fornisce il nome di una coda di messaggi non recapitabili, viene utilizzato, altrimenti il comportamento è NO.


USERID (ID utente)

Questo attributo specifica l'ID utente che deve essere utilizzato dall'MCA quando si tenta di avviare una sessione SNA sicura con un MCA remoto.

È possibile specificare un identificativo utente dell'attività di 20 caratteri.

Sull'estremità di ricezione, se le parole d'ordine vengono conservate in formato codificato e il software LU 6.2 utilizza un metodo di codifica differente, un tentativo di avviare il canale non riesce con dettagli di sicurezza non validi. È possibile evitare questo errore modificando la configurazione SNA di ricezione in:

- Disattivare la sostituzione della parola d'ordine oppure
- Definire un ID utente di sicurezza e una password.

 Su IBM MQ for z/OS, questo attributo è valido solo per canali di connessione client.

 Su altre piattaforme, questo attributo è valido per i tipi di canale:

- Mittente
- Server
- Richiedente
- Connessione client
- Mittente del cluster

XMITQ (Nome coda di trasmissione)

Questo attributo specifica il nome della coda di trasmissione da cui vengono richiamati i messaggi.

Fornire il nome della coda di trasmissione da associare a questo canale mittente o server, che corrisponde al gestore code sul lato opposto del canale. È possibile assegnare alla coda di trasmissione lo stesso nome del gestore code all'estremità remota.



Questo attributo è richiesto per i canali di tipo mittente o server e non è valido per altri tipi di canale.

Comandi e attributi del cluster IBM MQ

Esistono comandi cluster MQSC e PCF che è possibile utilizzare per aggiornare o reimpostare un cluster o per visualizzare, riprendere o sospendere un gestore code cluster. Inoltre, i comandi MQSC e PCF che definiscono canali, code e gestori code hanno attributi che si applicano ai cluster. Alcuni di questi attributi sono utilizzati dall'algoritmo di gestione del carico di lavoro del cluster.

Comandi MQSC

I comandi MQSC vengono visualizzati come vengono immessi dall'amministratore di sistema nella console di comandi. Non è necessario immettere i comandi in questo modo. Esistono diversi altri metodi, a seconda della piattaforma; ad esempio:

-  Su IBM MQ for IBM i, eseguire i comandi MQSC in modo interattivo dall'opzione 26 di **WRKMQM**. È anche possibile utilizzare i comandi CL oppure è possibile memorizzare i comandi MQSC in un file e utilizzare il comando CL **STRMQMMQSC**.
-  Su z/OS è possibile utilizzare la funzione COMMAND del programma di utilità **CSQUTIL**, le operazioni e i pannelli di controllo o la console z/OS.
- Su tutte le altre piattaforme, è possibile memorizzare i comandi in un file e utilizzare **runmqsc**.

In un comando MQSC, un nome cluster specificato utilizzando l'attributo CLUSTER può avere una lunghezza massima di 48 caratteri.

Un elenco di nomi cluster specificato utilizzando l'attributo CLUSNL può contenere fino a 256 nomi. Per creare un elenco nomi cluster, utilizzare il comando **DEFINE NAMELIST**.

IBM MQ Explorer

La GUI di IBM MQ Explorer può gestire un cluster con gestori code del repository su IBM WebSphere MQ for z/OS 6.0 o versioni successive. Non è necessario denominare un repository aggiuntivo su un sistema separato. Per versioni precedenti di IBM MQ for z/OS, IBM MQ Explorer non può gestire un cluster con gestori code del repository. È pertanto necessario denominare un repository aggiuntivo su un sistema che può essere gestito da IBM MQ Explorer.

Su IBM MQ for Windows e IBM MQ for Linux, puoi anche utilizzare IBM MQ Explorer per gestire i cluster. È anche possibile utilizzare il client IBM MQ Explorer autonomo.

Utilizzando IBM MQ Explorer, è possibile visualizzare le code cluster e richiedere informazioni sullo stato dei canali mittente cluster e ricevente cluster. IBM MQ Explorer include due procedure guidate, che è possibile utilizzare per eseguire le seguenti attività:

- Creare un cluster
- Unisci un gestore code indipendente a un cluster

Equivalenti PCF dei comandi MQSC specificamente per gestire i cluster

Comando MQSC	comando PCF equivalente
DISPLAY CLUSQMGR	MQCMD_INQUIRE_CLUSTER_Q_MGR
REFRESH CLUSTER	MQCMD_REFRESH_CLUSTER
RESET CLUSTER	MQCMD_RESET_CLUSTER
RESUME QMGR	MQCMD_RESUME_Q_MGR_CLUSTER
SUSPEND QMGR	MQCMD_SUSPEND_Q_MGR_CLUSTER

Informazioni correlate

[Cluster: utilizzo delle procedure consigliate per REFRESH CLUSTER](#)

Attributi cluster disponibili nei comandi di definizioni di canale

Attributi cluster che possono essere specificati nei comandi di definizione del canale.

I comandi DEFINE CHANNEL, ALTER CHANNEL e DISPLAY CHANNEL hanno due parametri specifici CHLTYPE per cluster: CLUSRCVR e CLUSSDR. Per definire un canale ricevente del cluster, utilizzare il comando DEFINE CHANNEL, specificando CHLTYPE (CLUSRCVR). Molti attributi su una definizione di canale ricevente del cluster sono gli stessi di quelli su una definizione di canale ricevente o mittente. Per definire un canale mittente del cluster, utilizzare il comando DEFINE CHANNEL, specificando CHLTYPE (CLUSSDR) e molti degli stessi attributi utilizzati per definire un canale mittente.

Non è più necessario specificare il nome del gestore code del repository completo quando si definisce un canale mittente del cluster. Se si conosce la convenzione di denominazione utilizzata per i canali nel proprio cluster, è possibile creare una definizione CLUSSDR utilizzando la costruzione +QMNAME+. La costruzione +QMNAME+ non è supportata su z/OS. Dopo la connessione IBM MQ modifica il nome del canale e sostituisce il corretto nome del gestore code del repository completo al posto di +QMNAME+. Il nome canale risultante viene troncato a 20 caratteri.

Per ulteriori informazioni sulle convenzioni di denominazione, consultare [Convenzioni di denominazione del cluster](#).

La tecnica funziona solo se la convenzione per la denominazione dei canali include il nome del gestore code. Ad esempio, si definisce un gestore code del repository completo denominato QM1 in un cluster denominato CLUSTER1 con un canale ricevente del cluster denominato CLUSTER1.QM1.ALPHA. Ogni altro gestore code può definire un canale mittente del cluster per questo gestore code utilizzando il nome del canale, CLUSTER1.+QMNAME+.ALPHA.

Se si utilizza la stessa convenzione di denominazione per tutti i canali, tenere presente che può essere presente una sola definizione +QMNAME+ alla volta.

I seguenti attributi nei comandi DEFINE CHANNEL e ALTER CHANNEL sono specifici per i canali cluster:

cluster

L'attributo CLUSTER specifica il nome del cluster a cui è associato questo canale. In alternativa, utilizzare l'attributo CLUSNL.

CLUSNL

L'attributo CLUSNL specifica un elenco nomi di nomi cluster.

NETPRTY

Solo ricevitori cluster.

L'attributo NETPRTY specifica una priorità di rete per il canale. NETPRTY aiuta le routine di gestione del carico di lavoro. Se esiste più di un possibile instradamento a una destinazione, la routine di gestione del carico di lavoro seleziona quello con la priorità più alta.

CLWLPRTY

Il parametro CLWLPRTY applica un fattore di priorità ai canali nella stessa destinazione per scopi di gestione del carico di lavoro. Questo parametro specifica la priorità del canale per la distribuzione del carico di lavoro del cluster. Il valore deve essere compreso nell'intervallo tra zero e 9, dove zero è la priorità più bassa e 9 è la più alta.

CLWLRANK

Il parametro CLWLRANK applica un fattore di classificazione a un canale per scopi di gestione del workload. Questo parametro specifica la classificazione di un canale ai fini della distribuzione del workload del cluster. Il valore deve essere compreso nell'intervallo tra zero e 9, dove zero è il rango più basso e 9 è il più alto.

CLWLWGHT

Il parametro CLWLWGHT applica un fattore di ponderazione a un canale per scopi di gestione del workload. CLWLWGHT pesa il canale in modo che sia possibile controllare la proporzione di messaggi inviati in quel canale. L'algoritmo del carico di lavoro del cluster utilizza CLWLWGHT per determinare la scelta della destinazione in modo che sia possibile inviare più messaggi su un determinato canale. Per impostazione predefinita tutti gli attributi di peso del canale sono lo stesso valore predefinito. L'attributo peso consente di allocare un canale su una macchina UNIX potente con un peso maggiore di un altro canale su un PC desktop di piccole dimensioni. Il peso maggiore indica che l'algoritmo del carico di lavoro del cluster seleziona la macchina UNIX più frequentemente del PC come destinazione per i messaggi.

CONNAME

Il valore CONNAME specificato in una definizione di canale ricevente del cluster viene utilizzato in tutto il cluster per identificare l'indirizzo di rete del gestore code. Fai attenzione a selezionare un valore per il parametro CONNAME che si risolve in tutto il tuo cluster IBM MQ. Non utilizzare un nome generico. Tenere presente che il valore specificato sul canale ricevente del cluster ha la precedenza su qualsiasi valore specificato in un canale mittente del cluster corrispondente.

Questi attributi sul comando DEFINE CHANNEL e sul comando ALTER CHANNEL si applicano anche al comando DISPLAY CHANNEL.

Nota: I canali mittenti del cluster definiti automaticamente prendono i loro attributi dalla corrispondente definizione di canale ricevente del cluster sul gestore code di ricezione. Anche se esiste un canale mittente del cluster definito manualmente, i suoi attributi vengono modificati automaticamente per garantire che corrispondano agli attributi nella definizione ricevente del cluster corrispondente. È possibile, ad esempio, definire un CLUSRCVR senza specificare un numero di porta nel parametro CONNAME, mentre si definisce manualmente un CLUSSDR che specifica un numero di porta. Quando il CLUSSDR definito automaticamente sostituisce quello definito manualmente, il numero porta (preso da CLUSRCVR) diventa vuoto. Verrà utilizzato il numero di porta predefinito e il canale avrà esito negativo.

Nota: Il comando DISPLAY CHANNEL non visualizza canali definiti automaticamente. Tuttavia, è possibile utilizzare il comando DISPLAY CLUSQMGR per esaminare gli attributi dei canali mittenti del cluster definiti automaticamente.

Utilizzare il comando DISPLAY CHSTATUS per visualizzare lo stato di un canale mittente cluster o ricevente cluster. Questo comando fornisce lo stato dei canali definiti manualmente e dei canali definiti automaticamente.

Le PCF equivalenti sono MQCMD_CHANGE_CHANNEL, MQCMD_COPY_CHANNEL, MQCMD_CREATE_CHANNEL e MQCMD_INQUIRE_CHANNEL.

Omissione del valore CONNAME in una definizione CLUSRCVR

In alcune circostanze, è possibile omettere il valore CONNAME su una definizione CLUSRCVR . Non devi omettere il valore CONNAME su z/OS.

Multi Su Multipiattaforme, il parametro del nome connessione TCP/IP di un canale ricevente del cluster è facoltativo. Se si lascia vuoto il nome connessione, IBM MQ genera un nome connessione per conto dell'utente, presumendo la porta predefinita e utilizzando l'indirizzo IP corrente del sistema. È possibile sovrascrivere il numero di porta predefinito, utilizzando comunque l'indirizzo IP corrente del sistema. Per ciascun nome di connessione, lasciare vuoto il nome IP e fornire il numero di porta tra parentesi; ad esempio:

(1415)

Il **CONNAME** generato è sempre nel formato decimale puntato (IPv4) o esadecimale (IPv6), invece che nel formato di nome host DNS alfanumerico.

Questa funzione è utile quando si dispone di macchine che utilizzano DHCP (Dynamic Host Configuration Protocol). Se non si fornisce un valore per CONNAME su un canale CLUSRCVR , non è necessario modificare la definizione CLUSRCVR . DHCP assegna un nuovo indirizzo IP.

Se si specifica uno spazio vuoto per CONNAME nella definizione CLUSRCVR , IBM MQ genera CONNAME dall'indirizzo IP del sistema. Solo il CONNAME generato viene memorizzato nei repository. Altri gestori code nel cluster non sanno che CONNAME era originariamente vuoto.

Se si immette il comando DISPLAY CLUSQMGR , viene visualizzato il CONNAME generato. Tuttavia, se si immette il comando DISPLAY CHANNEL dal gestore code locale, si nota che CONNAME è vuoto.

Se il gestore code viene arrestato e riavviato con un indirizzo IP differente, a causa di DHCP, IBM MQ rigenera il CONNAME e aggiorna i repository di conseguenza.

Concetti correlati

Bilanciamento del carico di lavoro nei cluster

Se un cluster contiene più di un'istanza della stessa coda, IBM MQ seleziona un gestore code a cui instradare un messaggio. Utilizza l'algoritmo di gestione del carico di lavoro del cluster e un certo numero di attributi specifici del carico di lavoro del cluster, per determinare il gestore code migliore da utilizzare.

z/OS Asynchronous behavior of CLUSTER commands on z/OS

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

Riferimenti correlati

Attributi cluster disponibili sui comandi di definizione della coda

Attributi del cluster che è possibile specificare nei comandi di definizione della coda.

Attributi cluster disponibili nei comandi di definizione del gestore code

Attributi cluster che è possibile specificare nei comandi di definizione del gestore code.

DISPLAY CLUSQMGR

Utilizzare il comando DISPLAY CLUSQMGR per visualizzare le informazioni sui gestori code in un cluster.

Aggiornamento cluster

Immettere il comando REFRESH CLUSTER da un gestore code per eliminare tutte le informazioni conservate localmente su un cluster. È improbabile che sia necessario utilizzare questo comando, tranne in circostanze eccezionali.

RESET CLUSTER: rimozione forzata di un gestore code da un cluster

Utilizzare il comando **RESET CLUSTER** per forzare la rimozione di un gestore code da un cluster in circostanze eccezionali.

SUSPEND QMGR, RESUME QMGR e cluster

Utilizzare il comando `SUSPEND QMGR` e `RESUME QMGR` per ridurre temporaneamente l'attività del cluster in entrata a questo gestore code, ad esempio, prima di eseguire la manutenzione su questo gestore code e quindi ripristinarla.

[“Attributi Bilanciamento carico di lavoro cluster - canale” a pagina 142](#)

Un elenco alfabetico degli attributi di canale utilizzati nel bilanciamento del carico di lavoro del cluster.

Attributi cluster disponibili sui comandi di definizione della coda

Attributi del cluster che è possibile specificare nei comandi di definizione della coda.

I comandi `DEFINE QLOCAL`, `DEFINE QREMOTE` e `DEFINE QALIAS`

Gli attributi del cluster nei comandi `DEFINE QLOCAL`, `DEFINE QREMOTE` e `DEFINE QALIAS` e i tre comandi `ALTER` equivalenti sono:

cluster

Specifica il nome del cluster a cui appartiene la coda.

CLUSNL

Specifica un elenco nomi di nomi cluster.

DEFBIND

Specifica il bind da utilizzare quando un'applicazione specifica `MQOO_BIND_AS_Q_DEF` sulla chiamata `MQOPEN`. Le opzioni per questo attributo sono:

- Specificare `DEFBIND(OPEN)` per collegare l'handle della coda a una specifica istanza della coda cluster quando la coda viene aperta. `DEFBIND(OPEN)` è il valore predefinito per questo attributo.
- Specificare `DEFBIND(NOTFIXED)` in modo che l'handle della coda non sia collegato ad alcuna istanza della coda cluster.
- Specificare `DEFBIND(GROUP)` per consentire a un'applicazione di richiedere che un gruppo di messaggi venga assegnato alla stessa istanza di destinazione.

Quando più code con lo stesso nome vengono pubblicizzate in un cluster del gestore code, le applicazioni possono scegliere se inviare tutti i messaggi da questa applicazione a una singola istanza (`MQOO_BIND_ON_OPEN`), per consentire all'algoritmo di gestione del carico di lavoro di selezionare la destinazione più adatta in base al messaggio (`MQOO_BIND_NOT_FIXED`) o consentire a un'applicazione di richiedere che un 'gruppo' di messaggi venga assegnato tutti alla stessa istanza di destinazione (`MQOO_BIND_ON_GROUP`). Il bilanciamento del carico di lavoro viene riconnesso tra gruppi di messaggi (senza richiedere `MQCLOSE` e `MQOPEN` della coda).

Quando si specifica `DEFBIND` su una definizione di coda, la coda viene definita con uno degli attributi, `MQBND_BIND_ON_OPEN`, `MQBND_BIND_NOT_FIXED` o `MQBND_BIND_ON_GROUP`. È necessario specificare `MQBND_BIND_ON_OPEN` o `MQBND_BIND_ON_GROUP` quando si utilizzano gruppi con cluster.

È necessario impostare l'attributo `DEFBIND` sullo stesso valore su tutte le istanze della stessa coda cluster.

CLWLRANK

Applica un fattore di classificazione a una coda per scopi di gestione del workload. Il parametro `CLWLRANK` non è supportato sulle code modello. L'algoritmo del carico di lavoro del cluster seleziona una coda di destinazione con la classificazione più alta. Per impostazione predefinita, `CLWLRANK` per tutte le code è impostato su zero.

Se la destinazione finale è un gestore code su un cluster diverso, è possibile impostare la classificazione di qualsiasi gestore code gateway intermedio all'intersezione dei cluster vicini. Con i gestori code intermedi classificati, l'algoritmo del carico di lavoro del cluster seleziona correttamente un gestore code di destinazione più vicino alla destinazione finale.

La stessa logica si applica alle code alias. La selezione della classificazione viene effettuata prima che venga controllato lo stato del canale e quindi anche i gestori code non accessibili sono disponibili per la selezione. Ciò ha l'effetto di consentire a un messaggio di essere instradato attraverso una rete, piuttosto che selezionarlo tra due destinazioni possibili (come la priorità). Quindi, se un canale non

viene avviato nel punto in cui la classificazione ha indicato, il messaggio non viene instradato alla successiva classificazione più alta, ma attende che un canale sia disponibile per quella destinazione (il messaggio è congelato nella coda di trasmissione).

CLWLPRTY

Applica un fattore di priorità ad una coda per scopi di gestione del workload. L'algoritmo del carico di lavoro del cluster seleziona una coda di destinazione con la priorità più alta. Per impostazione predefinita la priorità per tutte le code è impostata su zero.

Se sono presenti due possibili code di destinazione, è possibile utilizzare questo attributo per eseguire il failover di una destinazione sull'altra destinazione. La selezione della priorità viene effettuata una volta selezionato lo stato del canale. Tutti i messaggi vengono inviati alla coda con la priorità più alta, a meno che lo stato del canale verso quella destinazione non sia tanto favorevole quanto lo stato dei canali verso altre destinazioni. Ciò significa che solo le destinazioni più accessibili sono disponibili per la selezione. Ciò ha l'effetto di assegnare una priorità tra più destinazioni che sono tutte disponibili.

CLWLUSEQ

Specifica il comportamento di un'operazione MQPUT per una coda. Questo parametro specifica il comportamento di un'operazione MQPUT quando la coda di destinazione ha un'istanza locale e almeno un'istanza cluster remota (tranne dove MQPUT ha origine da un canale cluster). Questo parametro è valido solo per le code locali.

I valori possibili sono: QMGR (il funzionamento è quello specificato dal parametro CLWLUSEQ della definizione del gestore code), ANY (il gestore code tratta la coda locale come un'altra istanza della coda del cluster, ai fini della distribuzione del carico di lavoro), LOCAL (la coda locale è l'unica destinazione dell'operazione MQPUT, purché la coda locale sia abilitata all'inserimento). Il funzionamento di MQPUT dipende dall' algoritmo di gestione del carico di lavoro cluster.

I comandi DISPLAY QUEUE e DISPLAY QCLUSTER

Gli attributi nei comandi DEFINE QLOCAL, DEFINE QREMOTE e DEFINE QALIAS si applicano anche al comando DISPLAY QUEUE.

Per visualizzare le informazioni relative alle code cluster, specificare un tipo di coda QCLUSTER o la parola chiave CLUSINFO nel comando DISPLAY QUEUE oppure utilizzare il comando DISPLAY QCLUSTER.

Il comando DISPLAY QUEUE o DISPLAY QCLUSTER restituisce il nome del gestore code che ospita la coda (o i nomi di tutti i gestori code se è presente più di un'istanza della coda). Restituisce anche il nome di sistema per ogni gestore code che ospita la coda, il tipo di coda rappresentato e la data e ora in cui la definizione è diventata disponibile per il gestore code locale. Queste informazioni vengono restituite utilizzando gli attributi CLUSQMGR, QMID, CLUSQT, CLUSDATE e CLUSTIME.

Il nome del sistema per il gestore code (QMID) è un nome univoco generato dal sistema per il gestore code.

È possibile definire una coda cluster che sia anche una coda condivisa. Ad esempio, su z/OS è possibile definire:

```
DEFINE QLOCAL(MYQUEUE) CLUSTER(MYCLUSTER) QSGDISP(SHARED) CFSTRUCT(STRUCTURE)
```

Le PCF equivalenti sono MQCMD_CHANGE_Q, MQCMD_COPY_Q, MQCMD_CREATE_Q e MQCMD_INQUIRE_Q.

Concetti correlati

Bilanciamento del carico di lavoro nei cluster

Se un cluster contiene più di un'istanza della stessa coda, IBM MQ seleziona un gestore code a cui instradare un messaggio. Utilizza l'algoritmo di gestione del carico di lavoro del cluster e un certo numero di attributi specifici del carico di lavoro del cluster, per determinare il gestore code migliore da utilizzare.

 Asynchronous behavior of CLUSTER commands on z/OS

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

Riferimenti correlati

[Attributi cluster disponibili nei comandi di definizioni di canale](#)

[Attributi cluster che possono essere specificati nei comandi di definizione del canale.](#)

[Attributi cluster disponibili nei comandi di definizione del gestore code](#)

[Attributi cluster che è possibile specificare nei comandi di definizione del gestore code.](#)

DISPLAY CLUSQMGR

Utilizzare il comando DISPLAY CLUSQMGR per visualizzare le informazioni sui gestori code in un cluster.

Aggiornamento cluster

Immettere il comando REFRESH CLUSTER da un gestore code per eliminare tutte le informazioni conservate localmente su un cluster. È improbabile che sia necessario utilizzare questo comando, tranne in circostanze eccezionali.

RESET CLUSTER: rimozione forzata di un gestore code da un cluster

Utilizzare il comando **RESET CLUSTER** per forzare la rimozione di un gestore code da un cluster in circostanze eccezionali.

SUSPEND QMGR, RESUME QMGR e cluster

Utilizzare il comando SUSPEND QMGR e RESUME QMGR per ridurre temporaneamente l'attività del cluster in entrata a questo gestore code, ad esempio, prima di eseguire la manutenzione su questo gestore code e quindi ripristinarla.

[“Bilanciamento carico di lavoro cluster - Attributi coda” a pagina 144](#)

Un elenco alfabetico di attributi di coda utilizzati nel bilanciamento del carico di lavoro del cluster.

Attributi cluster disponibili nei comandi di definizione del gestore code

Attributi cluster che è possibile specificare nei comandi di definizione del gestore code.

Per specificare che un gestore code contiene un repository completo per un cluster, utilizzare il comando **ALTER QMGR** specificando l'attributo REPOS(*clustername*). Per specificare un elenco di diversi nomi cluster, definire un elenco nomi cluster e specificare l'attributo REPOSNL(*namelist*) sul comando **ALTER QMGR** :

```
DEFINE NAMELIST(CLUSTERLIST)
  DESCR('List of clusters whose repositories I host')
  NAMES(CLUS1, CLUS2, CLUS3)
ALTER QMGR REPOSNL(CLUSTERLIST)
```

È possibile fornire ulteriori attributi cluster sul comando ALTER QMGR

CLWLEXIT(*name*)

Specifica il nome di un'uscita utente da chiamare quando un messaggio viene inserito in una coda cluster.

CLWLDATA(*data*)

Specifica i dati da passare all'uscita utente del carico di lavoro del cluster.

CLWLLEN(*length*)

Specifica la quantità massima di dati del messaggio da trasmettere all'uscita utente del carico di lavoro cluster

CLWLMRUC(*channels*)

Specifica il numero massimo di canali cluster in uscita.

CLWLMRUC è un attributo del gestore code locale che non viene propagato intorno al cluster. Viene reso disponibile per le uscite del carico di lavoro del cluster e l'algoritmo del carico di lavoro del cluster che seleziona la destinazione per i messaggi.

CLWLUSEQ(LOCAL|ANY)

Specifica il comportamento di MQPUT quando la coda di destinazione dispone di un'istanza locale e di almeno un'istanza cluster remota. Se l'immissione ha origine da un canale cluster, questo attributo non viene applicato. È possibile specificare CLWLUSEQ sia come attributo della coda che come attributo del gestore code.

Se si specifica ANY, sia la coda locale che le code remote sono possibili destinazioni di MQPUT.

Se si specifica LOCAL, la coda locale è l'unica destinazione di MQPUT.

I PCF equivalenti sono MQCMD_CHANGE_Q_MGR e MQCMD_INQUIRE_Q_MGR.

Concetti correlati

Bilanciamento del carico di lavoro nei cluster

Se un cluster contiene più di un'istanza della stessa coda, IBM MQ seleziona un gestore code a cui instradare un messaggio. Utilizza l'algoritmo di gestione del carico di lavoro del cluster e un certo numero di attributi specifici del carico di lavoro del cluster, per determinare il gestore code migliore da utilizzare.

 Asynchronous behavior of CLUSTER commands on z/OS

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

Riferimenti correlati

Attributi cluster disponibili nei comandi di definizioni di canale

Attributi cluster che possono essere specificati nei comandi di definizione del canale.

Attributi cluster disponibili sui comandi di definizione della coda

Attributi del cluster che è possibile specificare nei comandi di definizione della coda.

DISPLAY CLUSQMGR

Utilizzare il comando DISPLAY CLUSQMGR per visualizzare le informazioni sui gestori code in un cluster.

Aggiornamento cluster

Immettere il comando REFRESH CLUSTER da un gestore code per eliminare tutte le informazioni conservate localmente su un cluster. È improbabile che sia necessario utilizzare questo comando, tranne in circostanze eccezionali.

RESET CLUSTER: rimozione forzata di un gestore code da un cluster

Utilizzare il comando **RESET CLUSTER** per forzare la rimozione di un gestore code da un cluster in circostanze eccezionali.

SUSPEND QMGR, RESUME QMGR e cluster

Utilizzare il comando SUSPEND QMGR e RESUME QMGR per ridurre temporaneamente l'attività del cluster in entrata a questo gestore code, ad esempio, prima di eseguire la manutenzione su questo gestore code e quindi ripristinarla.

“Bilanciamento carico di lavoro cluster - Attributi gestore code” a pagina 146

Un elenco alfabetico degli attributi del gestore code utilizzati nel bilanciamento del carico di lavoro del cluster.

DISPLAY CLUSQMGR

Utilizzare il comando DISPLAY CLUSQMGR per visualizzare le informazioni sui gestori code in un cluster.

Se si immette questo comando da un gestore code con un repository completo, le informazioni restituite si applicano ad ogni gestore code nel cluster. Altrimenti, le informazioni restituite si applicheranno solo ai gestori code interessati. Ossia, ogni gestore code a cui è stato tentato di inviare un messaggio e ogni gestore code che contiene un repository completo.

Le informazioni includono la maggior parte degli attributi del canale che si applicano ai canali mittente cluster e ricevente cluster. Inoltre, è possibile visualizzare i seguenti attributi:

CHANNEL

Il nome del canale ricevente del cluster per il gestore code.

DATAESCLUSIONI

La data in cui la definizione è diventata disponibile per il gestore code locale.

cluster

I cluster in cui si trova il gestore code.

CLUSTIME

La data in cui la definizione è divenuta disponibile per il gestore code locale.

DEFTYPE

Modalità di definizione del gestore code. DEFTYPE può essere uno dei seguenti valori:

CLUSDR

Un canale mittente del cluster è stato definito amministrativamente sul gestore code locale ma non ancora riconosciuto dal gestore code di destinazione. Per essere in questo stato, il gestore code locale ha definito un canale mittente del cluster manuale ma il gestore code di ricezione non ha accettato le informazioni sul cluster. Ciò può essere dovuto al fatto che il canale non è stato mai stabilito a causa della disponibilità o a un errore nella configurazione del mittente del cluster, ad esempio una mancata corrispondenza nella proprietà CLUSTER tra le definizioni del mittente e del destinatario. Si tratta di una condizione transitoria o di uno stato di errore che deve essere analizzato.

CLUSDRA

Questo valore rappresenta un gestore code cluster rilevato automaticamente, nessun canale mittente del cluster è definito localmente. Questo è il DEFTYPE per i gestori code del cluster per cui il gestore code locale non ha una configurazione locale ma ne è stato informato. Per esempio

- Se il gestore code locale è un gestore code del repository completo, deve essere il valore DEFTYPE per tutti i gestori code del repository parziale nel cluster.
- Se il gestore code locale è un repository parziale, potrebbe essere l'host di una coda del cluster utilizzata da questo gestore code locale o da un secondo gestore code del repository completo con cui questo gestore code è stato indicato di lavorare.

Se il valore DEFTYPE è CLUSDRA e i gestori code locali e remoti sono entrambi repository completi per il cluster indicato, la configurazione non è corretta in quanto è necessario definire un canale mittente del cluster definito localmente per convertirlo in un DEFTYPE di [CLUSDRB](#).

CLUSDRB

Un canale mittente del cluster è stato amministrativamente definito sul gestore code locale e accettato come canale cluster valido dal gestore code di destinazione. Questo è il DEFTYPE previsto di un gestore code del repository parziale configurato manualmente. Deve essere anche il DEFTYPE di qualsiasi CLUSQMGR da un repository completo ad un altro nel cluster. I canali mittenti del cluster manuali non devono essere configurati per repository parziali o da un gestore code del repository parziale a più di un repository completo. Se un DEFTYPE di CLUSSDRB viene visualizzato in una di queste situazioni, è necessario esaminarlo e correggerlo.

CLUSRCVR

Amministrativamente definito come un canale ricevente del cluster sul gestore code locale. Rappresenta il gestore code locale nel cluster.

Nota: Per identificare quali CLUSQMGRs sono gestori code repository completi per il cluster, consultare la proprietà [QMTYPE](#) .

Per ulteriori informazioni sulla definizione di canali cluster, consultare [Canali cluster](#).

QMTYPE

Se contiene un repository completo o solo un repository parziale.

STATO

Lo stato del canale mittente del cluster per questo gestore code.

SUSPEND

Indica se il gestore code è sospeso.

VERSIONE

La versione dell'installazione di IBM MQ a cui è associato il gestore code del cluster.

La versione ha il formato VVRRMMFF:

- VV: versione
- RR: release
- MM: livello di manutenzione
- FF: livello di correzione

XMITQ

La coda di trasmissione del cluster utilizzata dal gestore code.

Consultare anche il comando `DISPLAY QCLUSTER`. Ciò è brevemente descritto nella sezione [DISPLAY QUEUE](#) e nei comandi `DISPLAY QUEUE` e `DISPLAY QCLUSTER` di "Attributi cluster disponibili sui comandi di definizione della coda" a pagina 132. Per esempi di utilizzo di `DISPLAY QCLUSTER`, ricercare la serie di informazioni per "DISPLAY QCLUSTER" e "DIS QCLUSTER".

Concetti correlati

[Bilanciamento del carico di lavoro nei cluster](#)

Se un cluster contiene più di un'istanza della stessa coda, IBM MQ seleziona un gestore code a cui instradare un messaggio. Utilizza l'algoritmo di gestione del carico di lavoro del cluster e un certo numero di attributi specifici del carico di lavoro del cluster, per determinare il gestore code migliore da utilizzare.

 [Asynchronous behavior of CLUSTER commands on z/OS](#)

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

Riferimenti correlati

[Attributi cluster disponibili nei comandi di definizioni di canale](#)

[Attributi cluster che possono essere specificati nei comandi di definizione del canale.](#)

[Attributi cluster disponibili sui comandi di definizione della coda](#)

[Attributi del cluster che è possibile specificare nei comandi di definizione della coda.](#)

[Attributi cluster disponibili nei comandi di definizione del gestore code](#)

[Attributi cluster che è possibile specificare nei comandi di definizione del gestore code.](#)

[Aggiornamento cluster](#)

Immettere il comando `REFRESH CLUSTER` da un gestore code per eliminare tutte le informazioni conservate localmente su un cluster. È improbabile che sia necessario utilizzare questo comando, tranne in circostanze eccezionali.

[RESET CLUSTER: rimozione forzata di un gestore code da un cluster](#)

Utilizzare il comando **RESET CLUSTER** per forzare la rimozione di un gestore code da un cluster in circostanze eccezionali.

[SUSPEND QMGR, RESUME QMGR e cluster](#)

Utilizzare il comando `SUSPEND QMGR` e `RESUME QMGR` per ridurre temporaneamente l'attività del cluster in entrata a questo gestore code, ad esempio, prima di eseguire la manutenzione su questo gestore code e quindi ripristinarla.

Comando MQSC **`DISPLAY CLUSQMGR`**

Aggiornamento cluster

Immettere il comando `REFRESH CLUSTER` da un gestore code per eliminare tutte le informazioni conservate localmente su un cluster. È improbabile che sia necessario utilizzare questo comando, tranne in circostanze eccezionali.

Esistono tre forme di questo comando:

REFRESH CLUSTER(clustername) REPOS(NO)

Impostazione predefinita. Il gestore code conserva la conoscenza di tutti i gestori code cluster e code cluster definiti localmente e di tutti i gestori code cluster che sono repository completi. Inoltre, se il gestore code è un repository completo per il cluster, conserva anche la conoscenza degli altri gestori code del cluster. Tutto il resto viene rimosso dalla copia locale del repository e ricreato dagli altri repository completi nel cluster. I canali cluster non vengono arrestati se viene utilizzato `REPOS(NO)`. Un repository completo utilizza i canali `CLUSDR` per informare il resto del cluster che ha completato l'aggiornamento.

REFRESH CLUSTER(clustername) REPOS(YES)

Oltre al comportamento predefinito, vengono aggiornati anche gli oggetti che rappresentano gestori code del cluster del repository completo. Non è valido utilizzare questa opzione se il gestore code è un

repository completo, se utilizzato il comando avrà esito negativo con un errore AMQ9406/CSQX406E registrato. Se è un repository completo, è necessario prima modificarlo in modo che non sia un repository completo per il cluster in questione. L'ubicazione completa del repository viene ripristinata dalle definizioni CLUSSDR definite manualmente. Dopo l'aggiornamento con REPOS (YES) è stato emesso, il gestore code può essere modificato in modo che sia nuovamente un repository completo, se necessario.

REFRESH CLUSTER(*)


Aggiorna il gestore code in tutti i cluster di cui è membro. Se utilizzato con REPOS (YES) REFRESH CLUSTER(*) ha l'effetto aggiuntivo di forzare il gestore code a riavviare la ricerca di repository completi dalle informazioni nelle definizioni CLUSSDR locali. La ricerca viene eseguita anche se il canale CLUSSDR connette il gestore code a diversi cluster.

Nota: Per i cluster di grandi dimensioni, l'utilizzo del comando **REFRESH CLUSTER** può danneggiare il cluster mentre è in esecuzione e, di nuovo, a intervalli di 27 giorni, quando gli oggetti del cluster inviano automaticamente gli aggiornamenti di stato a tutti i gestori code interessati. Consultare [Refreshing in a large cluster can affect performance and availability of the cluster](#).

Concetti correlati

[Bilanciamento del carico di lavoro nei cluster](#)

Se un cluster contiene più di un'istanza della stessa coda, IBM MQ seleziona un gestore code a cui instradare un messaggio. Utilizza l'algoritmo di gestione del carico di lavoro del cluster e un certo numero di attributi specifici del carico di lavoro del cluster, per determinare il gestore code migliore da utilizzare.

 [Asynchronous behavior of CLUSTER commands on z/OS](#)

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

Riferimenti correlati

[Attributi cluster disponibili nei comandi di definizioni di canale](#)

Attributi cluster che possono essere specificati nei comandi di definizione del canale.

[Attributi cluster disponibili sui comandi di definizione della coda](#)

Attributi del cluster che è possibile specificare nei comandi di definizione della coda.

[Attributi cluster disponibili nei comandi di definizione del gestore code](#)

Attributi cluster che è possibile specificare nei comandi di definizione del gestore code.

[DISPLAY CLUSQMGR](#)

Utilizzare il comando DISPLAY CLUSQMGR per visualizzare le informazioni sui gestori code in un cluster.

[RESET CLUSTER: rimozione forzata di un gestore code da un cluster](#)

Utilizzare il comando **RESET CLUSTER** per forzare la rimozione di un gestore code da un cluster in circostanze eccezionali.

[SUSPEND QMGR, RESUME QMGR e cluster](#)

Utilizzare il comando SUSPEND QMGR e RESUME QMGR per ridurre temporaneamente l'attività del cluster in entrata a questo gestore code, ad esempio, prima di eseguire la manutenzione su questo gestore code e quindi ripristinarla.

Informazioni correlate

[Cluster: utilizzo delle procedure consigliate per REFRESH CLUSTER](#)

RESET CLUSTER: rimozione forzata di un gestore code da un cluster

Utilizzare il comando **RESET CLUSTER** per forzare la rimozione di un gestore code da un cluster in circostanze eccezionali.

È improbabile che sia necessario utilizzare questo comando, tranne in circostanze eccezionali.

È possibile immettere il comando **RESET CLUSTER** solo dai gestori code del repository completo. Il comando può assumere due forme, a seconda che si faccia riferimento al gestore code in base al nome o all'identificativo.

1. `RESET CLUSTER(clustername
) QMNAME(qmname) ACTION(FORCEREMOVE) QUEUES(NO)`

2. `RESET CLUSTER(clustername
) QMID(qmid) ACTION(FORCEREMOVE) QUEUES(NO)`

Non puoi specificare sia QMNAME che QMID. Se si utilizza QMNAMEe nel cluster è presente più di un gestore code con tale nome, il comando non viene eseguito. Utilizzare QMID invece di QMNAME per assicurarsi che il comando **RESET CLUSTER** venga eseguito.

Specificare QUEUES (NO) su un comando **RESET CLUSTER** è il valore predefinito. Specificando QUEUES (YES) si rimuovono i riferimenti alle code cluster di proprietà del gestore code dal cluster. I riferimenti vengono rimossi in aggiunta alla rimozione del gestore code dal cluster stesso.

I riferimenti vengono rimossi anche se il gestore code del cluster non è visibile nel cluster; forse perché è stato precedentemente rimosso in modo forzato, senza l'opzione QUEUES .

È possibile utilizzare il comando **RESET CLUSTER** se, ad esempio, un gestore code è stato eliminato ma dispone ancora di canali riceventi del cluster definiti per il cluster. Invece di attendere che IBM MQ rimuova queste definizioni (cosa che fa automaticamente), è possibile immettere il comando **RESET CLUSTER** per riordinare prima. Tutti gli altri gestori code nel cluster vengono informati che il gestore code non è più disponibile.

Se un gestore code è temporaneamente danneggiato, è possibile che si desideri informare gli altri gestori code nel cluster prima che tentino di inviargli i messaggi. **RESET CLUSTER** rimuove il gestore code danneggiato. Successivamente, quando il gestore code danneggiato funziona nuovamente, utilizzare il comando **REFRESH CLUSTER** per annullare l'effetto di **RESET CLUSTER** e restituire il gestore code al cluster. Se il gestore code si trova in un cluster di pubblicazione/sottoscrizione, sarà necessario reintegrare eventuali sottoscrizioni proxy richieste. Consultare [Considerazioni su REFRESH CLUSTER per i cluster di pubblicazione/sottoscrizione](#).

Nota: Per i cluster di grandi dimensioni, l'utilizzo del comando **REFRESH CLUSTER** può danneggiare il cluster mentre è in esecuzione e, di nuovo, a intervalli di 27 giorni, quando gli oggetti del cluster inviano automaticamente gli aggiornamenti di stato a tutti i gestori code interessati. Consultare [Refreshing in a large cluster can affect performance and availability of the cluster](#).

L'uso di **RESET CLUSTER** è l'unico modo per eliminare i canali mittenti del cluster definiti automaticamente.

Importante: Se il canale definito automaticamente da rimuovere è in dubbio, **RESET CLUSTER** non rimuove immediatamente tale canale. In questa situazione è necessario immettere un comando [RESOLVE CHANNEL](#) , prima del comando **RESET CLUSTER** .

È improbabile che questo comando sia necessario in circostanze normali. Il supporto IBM potrebbe consigliare di immettere il comando per riordinare le informazioni sul cluster contenute nei gestori code del cluster. Non utilizzare questo comando come scorciatoia per rimuovere un gestore code da un cluster. Il modo corretto per rimuovere un gestore code da un cluster è descritto in [Rimozione di un gestore code da un cluster](#).

Poiché i repository conservano le informazioni solo per 90 giorni, dopo tale periodo di tempo un gestore code rimosso in modo forzato può riconnettersi a un cluster. Si riconnette automaticamente, a meno che non sia stato eliminato. Se si desidera impedire a un gestore code di ricongiungersi a un cluster, è necessario adottare le misure di sicurezza appropriate.

Tutti i comandi cluster, tranne **DISPLAY CLUSQMR**, funzionano in modo asincrono. I comandi che modificano attributi oggetto che implicano il clustering aggiornano l'oggetto e inviano una richiesta al processore del repository. I comandi per gestire i cluster vengono controllati per la sintassi e viene inviata una richiesta al processore del repository.

Le richieste inviate al processore del repository vengono elaborate in modo asincrono, insieme alle richieste cluster ricevute da altri membri del cluster. L'elaborazione potrebbe richiedere molto tempo se devono essere propagate all'intero cluster per determinare se hanno esito positivo o meno.

Concetti correlati

Bilanciamento del carico di lavoro nei cluster

Se un cluster contiene più di un'istanza della stessa coda, IBM MQ seleziona un gestore code a cui instradare un messaggio. Utilizza l'algoritmo di gestione del carico di lavoro del cluster e un certo numero di attributi specifici del carico di lavoro del cluster, per determinare il gestore code migliore da utilizzare.

Asynchronous behavior of CLUSTER commands on z/OS

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

Riferimenti correlati

Attributi cluster disponibili nei comandi di definizioni di canale

Attributi cluster che possono essere specificati nei comandi di definizione del canale.

Attributi cluster disponibili sui comandi di definizione della coda

Attributi del cluster che è possibile specificare nei comandi di definizione della coda.

Attributi cluster disponibili nei comandi di definizione del gestore code

Attributi cluster che è possibile specificare nei comandi di definizione del gestore code.

DISPLAY CLUSQMGR

Utilizzare il comando DISPLAY CLUSQMGR per visualizzare le informazioni sui gestori code in un cluster.

Aggiornamento cluster

Immettere il comando REFRESH CLUSTER da un gestore code per eliminare tutte le informazioni conservate localmente su un cluster. È improbabile che sia necessario utilizzare questo comando, tranne in circostanze eccezionali.

SUSPEND QMGR, RESUME QMGR e cluster

Utilizzare il comando SUSPEND QMGR e RESUME QMGR per ridurre temporaneamente l'attività del cluster in entrata a questo gestore code, ad esempio, prima di eseguire la manutenzione su questo gestore code e quindi ripristinarla.

RESET CLUSTER (ripristino di un cluster)

SUSPEND QMGR, RESUME QMGR e cluster

Utilizzare il comando SUSPEND QMGR e RESUME QMGR per ridurre temporaneamente l'attività del cluster in entrata a questo gestore code, ad esempio, prima di eseguire la manutenzione su questo gestore code e quindi ripristinarla.

Mentre un gestore code è sospeso da un cluster, non riceve messaggi sulle code del cluster che ospita se esiste una coda disponibile con lo stesso nome su un gestore code alternativo nel cluster. Tuttavia, i messaggi che sono esplicitamente destinati a questo gestore code o dove la coda di destinazione è disponibile solo su questo gestore code, sono ancora indirizzati a questo gestore code.

La ricezione di ulteriori messaggi in entrata mentre il gestore code è sospeso può essere impedita arrestando i canali riceventi del cluster per questo cluster. Per arrestare i canali riceventi cluster per un cluster, utilizzare la modalità FORCE del comando SUSPEND QMGR.

Concetti correlati

Bilanciamento del carico di lavoro nei cluster

Se un cluster contiene più di un'istanza della stessa coda, IBM MQ seleziona un gestore code a cui instradare un messaggio. Utilizza l'algoritmo di gestione del carico di lavoro del cluster e un certo numero di attributi specifici del carico di lavoro del cluster, per determinare il gestore code migliore da utilizzare.

Asynchronous behavior of CLUSTER commands on z/OS

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

Attività correlate

Gestione di un gestore code

Riferimenti correlati

Attributi cluster disponibili nei comandi di definizioni di canale

Attributi cluster che possono essere specificati nei comandi di definizione del canale.

Attributi cluster disponibili sui comandi di definizione della coda

Attributi del cluster che è possibile specificare nei comandi di definizione della coda.

Attributi cluster disponibili nei comandi di definizione del gestore code

Attributi cluster che è possibile specificare nei comandi di definizione del gestore code.

DISPLAY CLUSQMGR

Utilizzare il comando `DISPLAY CLUSQMGR` per visualizzare le informazioni sui gestori code in un cluster.

Aggiornamento cluster

Immettere il comando `REFRESH CLUSTER` da un gestore code per eliminare tutte le informazioni conservate localmente su un cluster. È improbabile che sia necessario utilizzare questo comando, tranne in circostanze eccezionali.

RESET CLUSTER: rimozione forzata di un gestore code da un cluster

Utilizzare il comando **RESET CLUSTER** per forzare la rimozione di un gestore code da un cluster in circostanze eccezionali.

Gestore code in stato SUSPEND

RESUME QMGR

Bilanciamento del carico di lavoro nei cluster

Se un cluster contiene più di un'istanza della stessa coda, IBM MQ seleziona un gestore code a cui instradare un messaggio. Utilizza l'algoritmo di gestione del carico di lavoro del cluster e un certo numero di attributi specifici del carico di lavoro del cluster, per determinare il gestore code migliore da utilizzare.

Le destinazioni adatte vengono scelte dall'algoritmo di gestione del carico di lavoro del cluster, in base alla disponibilità del gestore code e della coda e in base a un numero di attributi specifici del carico di lavoro del cluster associati a canali, code e gestori code. Questi attributi sono descritti nei topic secondari.

Dopo aver configurato gli attributi specifici del carico di lavoro del cluster, se la configurazione non funziona come previsto, esplorare i dettagli su come l'algoritmo sceglie un gestore code. Consultare "L'algoritmo di gestione del workload del cluster" a pagina 147. Se i risultati di questo algoritmo non soddisfano le proprie esigenze, è possibile scrivere un programma di uscita utente del carico di lavoro del cluster e utilizzare questa uscita per instradare i messaggi alla coda di propria scelta nel cluster. Consultare Scrittura e compilazione delle uscite del carico di lavoro del cluster.

Nome attributo	Descrizione
Attributi canale	
<u>CLWLPRTY</u>	Specifica l'ordine di priorità per i canali per la distribuzione del workload del cluster.
<u>CLWLRANK</u>	Specifica la classificazione dei canali per la distribuzione del workload del cluster.
<u>CLWLWGHT</u>	Specifica il peso applicato ai canali CLUSSDR e CLUSRCVR per la distribuzione del carico di lavoro del cluster.
<u>NETPRTY</u>	Specifica la priorità per un canale CLUSRCVR .
Attributi Coda	
<u>CLWLPRTY</u>	Specifica la priorità delle code locali, remote o alias per la distribuzione del carico di lavoro del cluster.

Tabella 59. Riepilogo degli attributi specifici del carico di lavoro del cluster (Continua)

Nome attributo	Descrizione
<u>CLWLRANK</u>	Specifica la classificazione di una coda locale, remota o alias per la distribuzione del carico di lavoro del cluster.
<u>CLWLUSEQ</u>	Specifica se a un'istanza locale di una coda viene data preferenza come destinazione rispetto alle altre istanze in un cluster.
Attributi Gestore code	
<u>CLWLMRUC</u>	Imposta il numero di canali scelti più di recente.
<u>CLWLUSEQ</u>	Specifica se a un'istanza locale di una coda viene data la preferenza come destinazione rispetto ad altre istanze della coda in un cluster.

Concetti correlati

 [Asynchronous behavior of CLUSTER commands on z/OS](#)

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

Riferimenti correlati

[Attributi cluster disponibili nei comandi di definizioni di canale](#)

[Attributi cluster che possono essere specificati nei comandi di definizione del canale.](#)

[Attributi cluster disponibili sui comandi di definizione della coda](#)

[Attributi del cluster che è possibile specificare nei comandi di definizione della coda.](#)

[Attributi cluster disponibili nei comandi di definizione del gestore code](#)

[Attributi cluster che è possibile specificare nei comandi di definizione del gestore code.](#)

DISPLAY CLUSQMGR

Utilizzare il comando `DISPLAY CLUSQMGR` per visualizzare le informazioni sui gestori code in un cluster.

Aggiornamento cluster

Immettere il comando `REFRESH CLUSTER` da un gestore code per eliminare tutte le informazioni conservate localmente su un cluster. È improbabile che sia necessario utilizzare questo comando, tranne in circostanze eccezionali.

RESET CLUSTER: rimozione forzata di un gestore code da un cluster

Utilizzare il comando **RESET CLUSTER** per forzare la rimozione di un gestore code da un cluster in circostanze eccezionali.

SUSPEND QMGR, RESUME QMGR e cluster

Utilizzare il comando `SUSPEND QMGR` e `RESUME QMGR` per ridurre temporaneamente l'attività del cluster in entrata a questo gestore code, ad esempio, prima di eseguire la manutenzione su questo gestore code e quindi ripristinarla.

Attributi Bilanciamento carico di lavoro cluster - canale

Un elenco alfabetico degli attributi di canale utilizzati nel bilanciamento del carico di lavoro del cluster.

Nota: Specificare gli attributi del canale del carico di lavoro del cluster sui canali riceventi del cluster sui gestori code di destinazione. È probabile che qualsiasi bilanciamento specificato sui canali mittenti del cluster corrispondenti venga ignorato. Vedere [Canali cluster](#).

CLWLPRTY (Priorità carico di lavoro cluster)

L'attributo del canale CLWLPRTY specifica l'ordine di priorità per i canali per la distribuzione del workload del cluster. Il valore deve essere compreso nell'intervallo 0-9, dove 0 è la priorità più bassa e 9 è la più alta.

Utilizzare l'attributo del canale CLWLPRTY per impostare un ordine di priorità per le destinazioni cluster disponibili. IBM MQ seleziona le destinazioni con la priorità più alta prima di selezionare le destinazioni con la priorità di destinazione cluster più bassa. Se ci sono più destinazioni con la stessa priorità, seleziona la destinazione utilizzata meno di recente.

Se vi sono due destinazioni possibili, è possibile utilizzare questo attributo per consentire il failover. I messaggi vengono inviati al gestore code con il canale con priorità più alta. Se diventa non disponibile, i messaggi passano al successivo gestore code con priorità più alta. I gestori code con priorità più bassa agiscono come riserve.

IBM MQ controlla lo stato del canale prima di assegnare la priorità ai canali. Solo i gestori code sono candidati per la selezione.

Note:

- Specificare questo attributo sul canale ricevente del cluster sul gestore code di destinazione. È probabile che qualsiasi bilanciamento specificato sul canale mittente del cluster corrispondente venga ignorato. Consultare [Canali cluster](#).
- La disponibilità di un gestore code remoto si basa sullo stato del canale per tale gestore code. Quando i canali vengono avviati, il loro stato cambia diverse volte, con alcuni degli stati meno preferenziali rispetto all'algoritmo di gestione del workload del cluster. In pratica ciò significa che è possibile scegliere destinazioni con priorità più bassa (backup) mentre i canali verso destinazioni con priorità più alta (primaria) sono in fase di avvio.
- Se è necessario assicurarsi che nessun messaggio vada a una destinazione di backup, non utilizzare CLWLPRTY. Prendere in considerazione l'utilizzo di code separate o di CLWLRANK con un passaggio manuale dal primario al backup.

CLWLRANK (classificazione carico di lavoro cluster)

L'attributo del canale **CLWLRANK** specifica la classificazione dei canali per la distribuzione del workload del cluster. Il valore deve essere compreso nell'intervallo 0-9, dove 0 è il rango più basso e 9 è il più alto.

Utilizzare l'attributo del canale **CLWLRANK** se si desidera controllare la destinazione finale per i messaggi inviati a un gestore code in un altro cluster. Controllare la scelta della destinazione finale impostando la classificazione dei canali che collegano un gestore code ai gestori code gateway all'intersezione dei cluster.

Quando si imposta **CLWLRANK**, i messaggi prendono un instradamento specificato attraverso i cluster interconnessi verso una destinazione di livello superiore. Ad esempio, i messaggi arrivano a un gestore code del gateway che può inviarli a uno dei due gestori code utilizzando i canali 1 e 2. Vengono inviati automaticamente al gestore code connesso da un canale con la classificazione più elevata, in questo caso il canale al gestore code classificato 2.

IBM MQ ottiene la classificazione dei canali prima di controllarne lo stato. Ottenere la classificazione prima di controllare lo stato del canale significa che anche i canali non accessibili sono disponibili per la selezione. Consente l'instradamento dei messaggi attraverso la rete anche se la destinazione finale non è disponibile.

Note:

- Specificare questo attributo sul canale ricevente del cluster sul gestore code di destinazione. È probabile che qualsiasi bilanciamento specificato sul canale mittente del cluster corrispondente venga ignorato. Consultare [Canali cluster](#).
- Se è stato utilizzato anche l'attributo di priorità **CLWLPRTY**, IBM MQ seleziona tra le destinazioni disponibili. Se un canale non è disponibile per la destinazione con la classificazione più alta, il messaggio viene conservato nella coda di trasmissione. Viene rilasciato quando il canale diventa

disponibile. Il messaggio non viene inviato alla successiva destinazione disponibile nell'ordine di classificazione.

CLWLWGHT (Peso carico di lavoro cluster)

L'attributo di canale CLWLWGHT specifica il peso applicato ai canali CLUSSDR e CLUSRCVR per la distribuzione del carico di lavoro del cluster. Il valore deve essere compreso tra 1 e 99, dove 1 è il peso più basso e 99 è il più alto.

Utilizzare CLWLWGHT per inviare più messaggi ai server con maggiore potenza di elaborazione. Maggiore è il peso del canale, maggiore è il numero di messaggi inviati su tale canale.

Note:

- Specificare questo attributo sul canale ricevente del cluster sul gestore code di destinazione. È probabile che qualsiasi bilanciamento specificato sul canale mittente del cluster corrispondente venga ignorato. Consultare [Canali cluster](#).
- Quando CLWLWGHT viene modificato dal valore predefinito di 50 su qualsiasi canale, il bilanciamento del carico di lavoro dipende dal numero totale di volte in cui ogni canale è stato scelto per un messaggio inviato a una coda cluster. Per ulteriori informazioni, consultare [“L'algoritmo di gestione del workload del cluster”](#) a pagina 147.

NETPRTY (Priorità connessione di rete)

L'attributo del canale NETPRTY specifica la priorità per un canale CLUSRCVR . Il valore deve essere compreso nell'intervallo 0-9, dove 0 è la priorità più bassa e 9 è la più alta.

Utilizzare l'attributo NETPRTY per rendere una rete la rete principale e un'altra la rete di backup. Dato un insieme di canali ugualmente classificati, il clustering sceglie il percorso con la priorità più alta quando sono disponibili più percorsi.

Un esempio tipico di utilizzo dell'attributo del canale NETPRTY è quello di differenziare le reti che hanno costi o velocità differenti e connettono le stesse destinazioni.

Nota: Specificare questo attributo sul canale ricevente del cluster sul gestore code di destinazione. È probabile che qualsiasi bilanciamento specificato sul canale mittente del cluster corrispondente venga ignorato. Consultare [Canali cluster](#).

Concetti correlati

[L'algoritmo di gestione del workload del cluster](#)

L'algoritmo di gestione del carico di lavoro utilizza gli attributi di bilanciamento del workload e molte regole per selezionare la destinazione finale per i messaggi inseriti nelle code cluster.

Riferimenti correlati

[Bilanciamento carico di lavoro cluster - Attributi coda](#)

Un elenco alfabetico di attributi di coda utilizzati nel bilanciamento del carico di lavoro del cluster.

[Bilanciamento carico di lavoro cluster - Attributi gestore code](#)

Un elenco alfabetico degli attributi del gestore code utilizzati nel bilanciamento del carico di lavoro del cluster.

[“Attributi cluster disponibili nei comandi di definizioni di canale”](#) a pagina 129

Attributi cluster che possono essere specificati nei comandi di definizione del canale.

Bilanciamento carico di lavoro cluster - Attributi coda

Un elenco alfabetico di attributi di coda utilizzati nel bilanciamento del carico di lavoro del cluster.

CLWLPRTY

L'attributo della coda **CLWLPRTY** specifica la priorità delle code locali, remote o alias per la distribuzione del carico di lavoro del cluster. Il valore deve essere compreso nell'intervallo 0-9, dove 0 è la priorità più bassa e 9 è la più alta.

Utilizzare l'attributo coda **CLWLPRTY** per impostare una preferenza per code di destinazione. IBM MQ seleziona le destinazioni con la priorità più alta prima di selezionare le destinazioni con la priorità di destinazione cluster più bassa. Se ci sono più destinazioni con la stessa priorità, seleziona la destinazione utilizzata meno di recente.

IBM MQ ottiene la priorità dei gestori code dopo il controllo dello stato del canale. Solo i gestori code sono candidati per la selezione.

Nota:

La disponibilità di un gestore code remoto si basa sullo stato del canale per tale gestore code. Quando i canali vengono avviati, il loro stato cambia diverse volte, con alcuni degli stati meno preferenziali rispetto all'algoritmo di gestione del workload del cluster. In pratica ciò significa che è possibile scegliere destinazioni con priorità più bassa (backup) mentre i canali verso destinazioni con priorità più alta (primaria) sono in fase di avvio.

Se è necessario assicurarsi che nessun messaggio venga inviato a una destinazione di backup, non utilizzare **CLWLPRTY**. Prendere in considerazione l'utilizzo di code separate o **CLWLRANK** con un passaggio manuale dal primario al backup.

Se vi sono due destinazioni possibili, è possibile utilizzare questo attributo per consentire il failover. Il gestore code con priorità più alta riceve le richieste, i gestori code con priorità più bassa fungono da riserva. Se il gestore code con la priorità più alta ha esito negativo, prende il sopravvento il successivo gestore code con la priorità più alta disponibile.

CLWLRANK

L'attributo della coda **CLWLRANK** specifica la classificazione di una coda locale, remota o alias per la distribuzione del carico di lavoro del cluster. Il valore deve essere compreso nell'intervallo 0-9, dove 0 è il rango più basso e 9 è il più alto.

Utilizzare l'attributo della coda **CLWLRANK** se si desidera controllare la destinazione finale per i messaggi inviati a un gestore code in un altro cluster. Quando si imposta **CLWLRANK**, i messaggi prendono un instradamento specificato attraverso i cluster interconnessi verso una destinazione di livello superiore.

Ad esempio, è possibile che siano stati definiti due gestori code del gateway configurati in modo identico per migliorare la disponibilità di un gateway. Si supponga di aver definito le code alias del cluster sui gateway per una coda locale definita nel cluster. Se la coda locale diventa non disponibile, si intende congelare il messaggio su uno dei gateway in attesa che la coda diventi nuovamente disponibile. Per congelare la coda su un gateway, è necessario definire la coda locale con una classificazione più alta rispetto alle code alias del cluster sul gateway.

Se si definisce la coda locale con la stessa classificazione degli alias della coda e la coda locale non è disponibile, il messaggio viaggia tra i gateway. Quando si trova la coda locale non disponibile, il gestore code del primo gateway instrada il messaggio all'altro gateway. L'altro gateway tenta di consegnare nuovamente il messaggio alla coda locale di destinazione. Se la coda locale non è ancora disponibile, instrada nuovamente il messaggio al primo gateway. Il messaggio continua a essere spostato avanti e indietro tra i gateway finché la coda locale di destinazione non diventa nuovamente disponibile. Assegnando alla coda locale una classificazione più elevata, anche se la coda non è disponibile, il messaggio non viene reinstradato a una destinazione di livello inferiore.

IBM MQ ottiene la classificazione delle code prima di verificare lo stato del canale. Ottenere la classificazione prima di controllare lo stato del canale significa che anche le code non accessibili sono disponibili per la selezione. Consente l'instradamento dei messaggi attraverso la rete anche se la destinazione finale non è disponibile.

Se è stato utilizzato l'attributo di priorità IBM MQ seleziona tra le destinazioni disponibili. Se un canale non è disponibile per la destinazione con la classificazione più alta, il messaggio viene conservato nella coda di trasmissione. Viene rilasciato quando il canale diventa disponibile. Il messaggio non viene inviato alla successiva destinazione disponibile nell'ordine di classificazione.

CLWLUSEQ

L'attributo della coda **CLWLUSEQ** specifica se ad una istanza locale di una coda viene data la preferenza come destinazione rispetto ad altre istanze in un cluster.

L'attributo della coda **CLWLUSEQ** è valido solo per code locali. Si applica solo se il messaggio viene inserito da un'applicazione o da un canale che non è un canale cluster.

LOCALE

La coda locale è l'unica destinazione di MQPUT, purché la coda locale sia abilitata. Il funzionamento di MQPUT dipende dalla [gestione del workload del cluster](#).

QMGR

Il comportamento è quello specificato dall'attributo gestore code **CLWLUSEQ**.

ANY

MQPUT considera la coda locale come qualsiasi altra istanza della coda nel cluster per la distribuzione del carico di lavoro.

Concetti correlati

[L'algoritmo di gestione del workload del cluster](#)

L'algoritmo di gestione del carico di lavoro utilizza gli attributi di bilanciamento del workload e molte regole per selezionare la destinazione finale per i messaggi inseriti nelle code cluster.

Riferimenti correlati

[Attributi Bilanciamento carico di lavoro cluster - canale](#)

Un elenco alfabetico degli attributi di canale utilizzati nel bilanciamento del carico di lavoro del cluster.

[Bilanciamento carico di lavoro cluster - Attributi gestore code](#)

Un elenco alfabetico degli attributi del gestore code utilizzati nel bilanciamento del carico di lavoro del cluster.

["Attributi cluster disponibili sui comandi di definizione della coda" a pagina 132](#)

Attributi del cluster che è possibile specificare nei comandi di definizione della coda.

Bilanciamento carico di lavoro cluster - Attributi gestore code

Un elenco alfabetico degli attributi del gestore code utilizzati nel bilanciamento del carico di lavoro del cluster.

CLWLMRUC

L'attributo Gestore code **CLWLMRUC** imposta il numero di canali scelti più di recente. L'algoritmo di gestione del carico di lavoro del cluster utilizza **CLWLMRUC** per limitare il numero di canali cluster in uscita attivi. Il valore deve essere compreso tra 1 e 999 999 999.

Il valore predefinito iniziale è 999 999 999.

CLWLUSEQ

L'attributo gestore code **CLWLUSEQ** specifica se a un'istanza locale di una coda viene data la preferenza come destinazione rispetto ad altre istanze della coda in un cluster. L'attributo si applica se l'attributo della coda **CLWLUSEQ** è impostato su QMGR.

L'attributo della coda **CLWLUSEQ** è valido solo per code locali. Si applica solo se il messaggio viene inserito da un'applicazione o da un canale che non è un canale cluster.

LOCALE

La coda locale è l'unica destinazione di MQPUT. LOCAL è il valore predefinito.

ANY

MQPUT considera la coda locale come qualsiasi altra istanza della coda nel cluster per la distribuzione del carico di lavoro.

Concetti correlati

L' algoritmo di gestione del workload del cluster

L' algoritmo di gestione del carico di lavoro utilizza gli attributi di bilanciamento del workload e molte regole per selezionare la destinazione finale per i messaggi inseriti nelle code cluster.

Riferimenti correlati

Attributi Bilanciamento carico di lavoro cluster - canale

Un elenco alfabetico degli attributi di canale utilizzati nel bilanciamento del carico di lavoro del cluster.

Bilanciamento carico di lavoro cluster - Attributi coda

Un elenco alfabetico di attributi di coda utilizzati nel bilanciamento del carico di lavoro del cluster.

“Attributi cluster disponibili nei comandi di definizione del gestore code” a pagina 134

Attributi cluster che è possibile specificare nei comandi di definizione del gestore code.

L'algoritmo di gestione del workload del cluster

L' algoritmo di gestione del carico di lavoro utilizza gli attributi di bilanciamento del workload e molte regole per selezionare la destinazione finale per i messaggi inseriti nelle code cluster.

L' algoritmo di gestione del carico di lavoro viene esercitato ogni volta che è richiesta una scelta di destinazione:

- Viene utilizzato nel punto in cui viene aperta una coda cluster, utilizzando l'opzione MQOO_BIND_ON_OPEN .
- Viene utilizzato ogni volta che un messaggio viene inserito in una coda cluster quando viene aperto con MQOO_BIND_NOT_FIXED.
- Viene utilizzato ogni volta che viene avviato un nuovo gruppo di messaggi quando MQOO_BIND_ON_GROUP viene utilizzato per aprire una coda cluster.
- Per l' instradamento host argomento, viene utilizzato ogni volta che un messaggio viene pubblicato in un argomento con cluster. Se il gestore code locale non è un host per questo argomento, l' algoritmo viene utilizzato per scegliere un gestore code host attraverso cui instradare il messaggio.

La seguente sezione descrive l' algoritmo di gestione del carico di lavoro utilizzato quando si determina la destinazione finale per i messaggi inseriti nelle code cluster. Queste regole sono influenzate dalle impostazioni applicate ai seguenti attributi per code, gestori code e canali:

Code	Gestori code	Canali
<ul style="list-style-type: none">• CLWLPRTY¹• CLWLRANK¹• CLWLUSEQ¹• PUT / PUB	<ul style="list-style-type: none">• CLWLMRUC• CLWLUSEQ¹	<ul style="list-style-type: none">• CLWLPRTY• CLWLRANK• CLWLWGHT• NETPRTY

Inizialmente, il gestore code crea un elenco di destinazioni possibili da due procedure:

- Corrispondenza tra `ObjectName` e `ObjectQmgrName` di destinazione e le definizioni alias del gestore code condivise negli stessi cluster del gestore code.
- Ricerca di instradamenti univoci (ovvero canali) a un gestore code che ospita una coda con il nome `ObjectName` e che si trova in uno dei cluster di cui il gestore code è membro.

L' algoritmo passa attraverso le seguenti regole per eliminare le destinazioni dall'elenco di destinazioni possibili.

1. Le istanze remote di code o argomenti o i canali CLUSRCVR remoti che non condividono un cluster con il gestore code locale vengono eliminati.

¹ Questo attributi si applica solo quando si sceglie una coda cluster, non quando si sceglie un argomento.

2. Se viene specificato un nome di coda o di argomento, i canali CLUSRCVR remoti che non si trovano nello stesso cluster della coda o dell'argomento vengono eliminati.

Nota: Tutte le code, gli argomenti e i canali rimanenti in questa fase vengono resi disponibili per l'uscita del carico di lavoro del cluster, se è configurata.
3. Tutti i canali per i gestori code o gli alias dei gestori code che hanno un valore CLWLRANK inferiore alla classificazione massima di tutti i canali rimanenti o degli alias dei gestori code vengono eliminati.
4. Vengono eliminate tutte le code (non gli alias del gestore code) con CLWLRANK inferiore alla classificazione massima di tutte le code rimanenti.
5. Se rimane più di un'istanza di una coda, di un argomento o di un alias del gestore code e se è abilitata la pubblicazione, tutte le istanze disabilitate vengono eliminate.

Nota: Se rimangono solo le istanze disabilitate di inserimento, solo le operazioni di interrogazione avranno esito positivo, tutte le altre operazioni avranno esito negativo con MQRC_CLUSTER_PUT_INITED.
6. Quando si sceglie una coda, se la serie risultante di code contiene l'istanza locale della coda, di solito viene utilizzata l'istanza locale. L'istanza locale della coda viene utilizzata se si verifica una delle seguenti condizioni:
 - L'attributo use - queue della coda, CLWLUSEQ, è impostato su LOCAL.
 - Entrambe le seguenti istruzioni sono vere:
 - L'attributo use - queue della coda, CLWLUSEQ, è impostato su QMGR.
 - L'attributo use - queue del gestore code, CLWLUSEQ, è impostato su LOCAL.
 - Il messaggio viene ricevuto su un canale cluster piuttosto che essere inserito da una applicazione locale.
 - Per le code definite localmente che sono definite con CLWLUSEQ (ANY) o che ereditano la stessa impostazione dal gestore code, i seguenti punti sono veri, all'interno della più ampia serie di condizioni che si applicano:
 - La coda locale viene scelta in base al status dei canali CLUSRCVR definiti localmente nello stesso cluster della coda. Questo stato viene confrontato con lo stato dei canali CLUSSDR che portano il messaggio a code definite in remoto con lo stesso nome.

Ad esempio, esiste un CLUSRCVR nello stesso cluster della coda. Tale CLUSRCVR ha lo stato STOPPING, mentre le altre code con lo stesso nome nel cluster hanno lo stato RUNNING o INACTIVE. In questo caso, verranno scelti i canali remoti e non verranno utilizzati i canali CLUSSDR locali.
 - La coda locale viene scelta in base al numero dei canali CLUSRCVR, in qualsiasi confronto con i canali CLUSSDR dello stesso stato, che portano il messaggio a code definite in remoto con lo stesso nome.

Ad esempio, vi sono quattro canali CLUSRCVR nello stesso cluster della coda e un canale CLUSSDR. Tutti i canali hanno lo stesso stato di INATTIVO o IN ESECUZIONE. Pertanto, ci sono cinque canali tra cui scegliere e due istanze della coda. Quattro quinti (80%) dei messaggi vanno alla coda locale.
7. Se rimane più di un gestore code, se uno di essi non è sospeso, tutti quelli sospesi vengono eliminati.
8. Se rimane più di un'istanza remota di una coda o di un argomento, vengono inclusi tutti i canali inattivi o in esecuzione. Le costanti di stato sono elencate:
 - MQCHS_INATTIVO
 - MQCHS_RUNNING
9. Se non rimane alcuna istanza remota di una coda o di un argomento, vengono inclusi tutti i canali in stato di bind, inizializzazione, avvio o arresto. Le costanti di stato sono elencate:
 - MQCHS_BINDING
 - MQCHS_INITIALIZING

- MQCHS_STARTING
 - MQCHS_STOPPING
10. Se non rimane alcuna istanza remota di una coda o di un argomento, vengono inclusi tutti i canali che vengono tentati di nuovo. La costante di stato è elencata:
- MQCHS_RETRYING
11. Se non rimane alcuna istanza remota di una coda o di un argomento, vengono inclusi tutti i canali in stato di richiesta, in pausa o arrestato. Le costanti di stato sono elencate:
- MQCHS_XX_ENCODE_CASE_ONE richiesta
 - MQCHS_PAUSED
 - MQCHS_STOPPED
 - MQCH_SWITCHING
12. Se rimangono più istanze remote di una coda o di un argomento su qualsiasi gestore code, vengono scelti i canali con il valore NETPRTY più elevato per ciascun gestore code.
13. Tutti i canali rimanenti e gli alias del gestore code diversi dai canali e gli alias con la massima priorità, CLWLPRTY, vengono eliminati. Se rimangono degli alias del gestore code, vengono conservati i canali per il gestore code.
14. Se si sta scegliendo una coda:
- Tutte le code diverse da quelle con la priorità più alta, CLWLPRTY, vengono eliminate e i canali vengono mantenuti.
15. I restanti canali vengono quindi ridotti a non più del numero massimo consentito di canali utilizzati più recentemente, CLWLMRUC, eliminando i canali con i valori più bassi di MQWDR DestSeqNumber.

Nota: I messaggi di controllo del cluster interni vengono inviati utilizzando lo stesso algoritmo del carico di lavoro del cluster, se appropriato.

Una volta calcolato l'elenco di destinazioni valide, i messaggi vengono bilanciati sul carico di lavoro, utilizzando la seguente logica:

- Quando più di un'istanza remota di una destinazione rimane e tutti i canali di quella destinazione hanno CLWLWGHT impostato sull'impostazione predefinita di 50, viene scelto il canale utilizzato meno di recente. Ciò equivale approssimativamente a uno stile round-robin di bilanciamento del carico di lavoro quando esistono più istanze remote.
- Quando più di un'istanza remota di una destinazione rimane e uno o più canali di tali code hanno CLWLWGHT impostato su un'impostazione non predefinita (anche se tutti hanno un valore non predefinito corrispondente), l'instradamento diventa dipendente dalle ponderazioni relative di ciascun canale e dal numero totale di volte in cui ciascun canale è stato precedentemente scelto durante l'invio dei messaggi.
- Quando si osserva la distribuzione dei messaggi per una singola coda cluster con più istanze, ciò può portare a una distribuzione non bilanciata in un sottoinsieme di istanze della coda. Questo perché è l'utilizzo cronologico di ogni canale mittente del cluster da questo gestore code che viene bilanciato, non solo il traffico di messaggi per tale coda. Se questo comportamento non è desiderato, completare una delle seguenti operazioni:
 - Impostare CLWLWGHT su 50 su tutti i canali riceventi del cluster se è richiesta una distribuzione uniforme.
 - Oppure, se alcune istanze della coda devono essere pesate in modo diverso rispetto ad altre, definire tali code in un cluster dedicato, con canali riceventi del cluster dedicati definiti. Questa azione isola il bilanciamento del carico di lavoro di tali code da altre nel cluster.
- I dati cronologici utilizzati per bilanciare i canali vengono reimpostati se gli attributi del carico di lavoro del cluster dei canali riceventi del cluster disponibili vengono modificati o se lo stato di un canale ricevente del cluster diventa disponibile. La modifica agli attributi del carico di lavoro dei canali mittenti del cluster definiti manualmente non reimposta i dati cronologici.

- Quando si considera la logica di uscita del carico di lavoro del cluster, il canale scelto è quello con l'MQWDR.MQWDR.DestSeqFactor. Ogni volta che viene scelto un canale, questo valore viene aumentato di circa 1000/CLWLWGHT. Se è presente più di un canale con il valore più basso, uno dei canali con il valore più basso di MQWDR.DestSeqNumber.

La distribuzione dei messaggi dell'utente non è sempre esatta perché la gestione e la manutenzione del cluster fanno sì che i messaggi fluiscano attraverso i canali. Il risultato è una distribuzione non uniforme dei messaggi utente che può richiedere del tempo per stabilizzarsi. A causa della combinazione di messaggi di amministrazione e utente, non fare affidamento sulla distribuzione esatta dei messaggi durante il bilanciamento del carico di lavoro.

Riferimenti correlati

Attributi Bilanciamento carico di lavoro cluster - canale

Un elenco alfabetico degli attributi di canale utilizzati nel bilanciamento del carico di lavoro del cluster.

Bilanciamento carico di lavoro cluster - Attributi coda

Un elenco alfabetico di attributi di coda utilizzati nel bilanciamento del carico di lavoro del cluster.

Bilanciamento carico di lavoro cluster - Attributi gestore code

Un elenco alfabetico degli attributi del gestore code utilizzati nel bilanciamento del carico di lavoro del cluster.

Asynchronous behavior of CLUSTER commands on z/OS

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

For both REFRESH CLUSTER and RESET CLUSTER, message CSQM130I is sent to the command issuer indicating that a request has been sent. This message is followed by message CSQ9022I to indicate that the command has completed successfully, in that a request has been sent. It does not indicate that the cluster request has been completed successfully.

Any errors are reported to the z/OS console on the system where the channel initiator is running, they are not sent to the command issuer.

The asynchronous behavior is in contrast to CHANNEL commands. A message indicating that a channel command has been accepted is issued immediately. At some later time, when the command has been completed, a message indicating either normal or abnormal completion is sent to the command issuer.

Related concepts

Bilanciamento del carico di lavoro nei cluster

Se un cluster contiene più di un'istanza della stessa coda, IBM MQ seleziona un gestore code a cui instradare un messaggio. Utilizza l'algoritmo di gestione del carico di lavoro del cluster e un certo numero di attributi specifici del carico di lavoro del cluster, per determinare il gestore code migliore da utilizzare.

Related tasks

Checking that async commands for distributed networks have finished

Related reference

Attributi cluster disponibili nei comandi di definizioni di canale

Attributi cluster che possono essere specificati nei comandi di definizione del canale.

Attributi cluster disponibili sui comandi di definizione della coda

Attributi del cluster che è possibile specificare nei comandi di definizione della coda.

Attributi cluster disponibili nei comandi di definizione del gestore code

Attributi cluster che è possibile specificare nei comandi di definizione del gestore code.

DISPLAY CLUSQMGR

Utilizzare il comando DISPLAY CLUSQMGR per visualizzare le informazioni sui gestori code in un cluster.

Aggiornamento cluster

Immettere il comando REFRESH CLUSTER da un gestore code per eliminare tutte le informazioni conservate localmente su un cluster. È improbabile che sia necessario utilizzare questo comando, tranne in circostanze eccezionali.

RESET CLUSTER: rimozione forzata di un gestore code da un cluster

Utilizzare il comando **RESET CLUSTER** per forzare la rimozione di un gestore code da un cluster in circostanze eccezionali.

SUSPEND QMGR, RESUME QMGR e cluster

Utilizzare il comando **SUSPEND QMGR** e **RESUME QMGR** per ridurre temporaneamente l'attività del cluster in entrata a questo gestore code, ad esempio, prima di eseguire la manutenzione su questo gestore code e quindi ripristinarla.

Programmi canale

Questa sezione esamina i diversi tipi di programmi di canale (MCA) disponibili per l'uso presso i canali.

I nomi degli MCA vengono mostrati nelle tabelle seguenti.

Nome programma	Direzione di connessione	Comunicazione
amqrmppa		Qualsiasi
runmqlsr	In entrata	Qualsiasi
amqcrs6a	In entrata	LU 6.2
amqcrsta	In entrata	TCP
runmqchl	In uscita	Qualsiasi
runmqchi	In uscita	Qualsiasi

runmqlsr (Esegui listener IBM MQ), runmqchl (Esegui canale IBM MQ) e runmqchi (Esegui iniziatore di canale IBM MQ) sono comandi di controllo che è possibile immettere sulla riga comandi.

amqcrsta viene richiamato per i canali TCP sui sistemi AIX and Linux che utilizzano inetd, dove non viene avviato alcun listener.

amqcrs6a viene richiamata come programma di transazione quando si utilizza LU6.2

IBM i Lavori di intercomunicazione su IBM i

I seguenti lavori sono associati a Intercommunication su IBM i. I nomi sono contenuti nella seguente tabella.

Nome lavoro	Descrizione
AMQCLMAA	Discriminatore non sudd. in thread
AMQCRSTA	Lavoro risp. non sudd. in thread
AMQRMPPA	Job Pool di Canale
RUNMQCHI	Inizializzatore di Canale
RUNMQCHL	Job di Canale
RUNMQLSR	Discriminatore suddiviso in thread

IBM i Stati del canale su IBM i

Gli stati dei canali vengono visualizzati nel pannello Gestione canali

Tabella 63. Stati del canale su IBM i

Nome stato	Significato
IN AVVIO	Il canale è pronto per iniziare la negoziazione con l'MCA di destinazione
Binding	Istituzione di una sessione e scambio di dati iniziale
In fase di richiesta	Canale richiedente che inizia una connessione
IN ESECUZIONE	Trasferimento o pronto per il trasferimento
INTERROTTO	In attesa dell'intervallo di tentativi del messaggio
IN FASE DI ARRESTO	Stabilire se riprovare o arrestare
Nuovo tentativo in corso	In attesa del tentativo successivo
ARRESTATO	Canale arrestato a causa di un errore o perché è stato immesso un comando end - channel
INATTIVO	Il canale ha terminato l'elaborazione normalmente o il canale non è mai stato avviato
*Nessuno	Nessuno stato (solo per canali di connessione server)

ALW Esempio: pianificazione di un canale di messaggi in AIX, Linux, and Windows

Queste informazioni forniscono un esempio dettagliato di come collegare due gestori code in modo che i messaggi possano essere inviati tra loro.

Informazioni su questa attività

In tutti gli esempi, i comandi MQSC vengono visualizzati come vengono visualizzati in un file di comandi e come vengono immessi nella riga comandi. I due metodi sembrano identici, ma, per immettere un comando dalla riga comandi, è necessario prima immettere `runmqsc`, per il gestore code predefinito o `runmqsc qmname` dove `qmname` è il nome del gestore code richiesto. Quindi immettere qualsiasi numero di comandi, come mostrato negli esempi.

Un metodo alternativo consiste nel creare un file contenente questi comandi. Eventuali errori nei comandi sono quindi facili da correggere. Se è stato richiamato il file `mqsc.in`, per eseguirlo sul gestore code QMNAME utilizzare:

```
runmqsc QMNAME < mqsc.in > mqsc.out
```

È possibile verificare i comandi nel file prima di eseguirlo utilizzando:

```
runmqsc -v QMNAME < mqsc.in > mqsc.out
```

Per la portabilità, è necessario limitare la lunghezza della linea dei comandi a 72 caratteri. Utilizzare un carattere di concatenazione per continuare su più di una riga. Su Windows utilizzare Ctrl - z per terminare l'input sulla riga comandi. Sui sistemi AIX and Linux utilizzare Ctrl - d. In alternativa, utilizzare il comando **end**.

Figura 7 a pagina 153 mostra lo scenario di esempio.

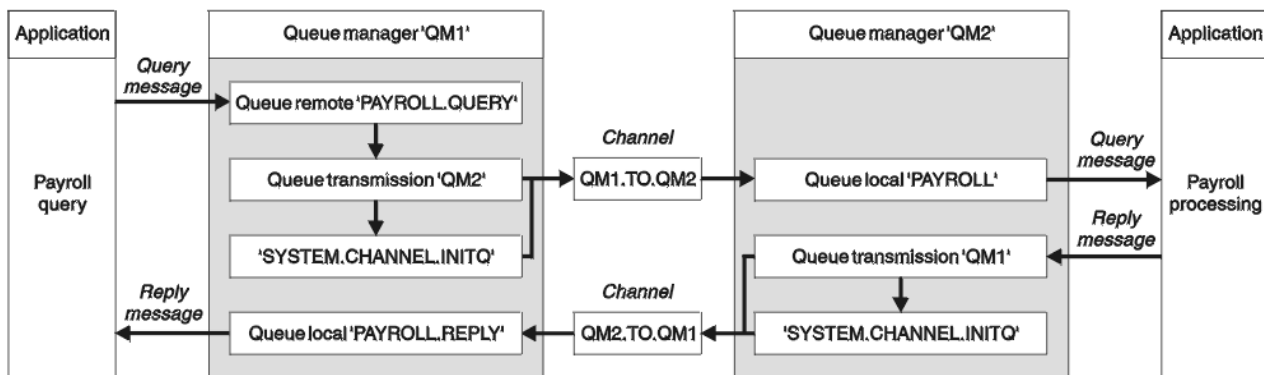


Figura 7. Esempio di canale dei messaggi per sistemi AIX, Linux, and Windows

L'esempio riguarda un'applicazione di query payroll connessa al gestore code QM1 che invia messaggi di query payroll a un'applicazione di elaborazione payroll in esecuzione sul gestore code QM2. L'applicazione di query del libro paga ha bisogno delle risposte alle sue query inviate a QM1. I messaggi di query payroll vengono inviati da QM1 a QM2 su un canale mittente - destinatario denominato QM1.TO.QM2 e i messaggi di risposta vengono inviati di nuovo da QM2 a QM1 su un altro canale mittente - destinatario denominato QM2.TO.QM1. Entrambi questi canali vengono attivati per essere avviati non appena hanno un messaggio da inviare all'altro gestore code.

L'applicazione di query payroll inserisce un messaggio di query nella coda remota "PAYROLL.QUERY" definito in QM1. Questa definizione di coda remota viene risolta nella coda locale "PAYROLL" su QM2. Inoltre, l'applicazione della query payroll specifica che la risposta alla query viene inviata alla coda locale "PAYROLL.REPLY" su QM1. L'applicazione di elaborazione del libro paga richiama i messaggi dalla coda locale "PAYROLL" su QM2 e invia le risposte ovunque siano richieste; in questo caso, la coda locale "PAYROLL.REPLY" su QM1.

Nelle definizioni di esempio per TCP/IP, QM1 ha un indirizzo host 192.0.2.0 ed è in ascolto sulla porta 1411 e QM2 ha un indirizzo host 192.0.2.1 ed è in ascolto sulla porta 1412. L'esempio presuppone che questi siano già definiti sul sistema e disponibili per l'uso.

Le definizioni oggetto che devono essere create su QM1 sono:

- Definizione coda remota, PAYROLL.QUERY
- Definizione della coda di trasmissione, QM2 (valore predefinito = nome gestore code remoto)
- Definizione del canale mittente, QM1.TO.QM2
- Definizione canale ricevente, QM2.TO.QM1
- Definizione coda di risposta, PAYROLL.REPLY

Le definizioni di oggetto che devono essere create su QM2 sono:

- Definizione coda locale, PAYROLL
- Definizione coda di trasmissione, QM1 (valore predefinito=nome gestore code remoto)
- Definizione canale mittente, QM2.TO.QM1
- Definizione di canale destinatario, QM1.TO.QM2

I dettagli di collegamento vengono forniti nell'attributo CONNAME delle definizioni del canale mittente.

È possibile visualizzare un diagramma della disposizione in [Figura 7 a pagina 153](#).

Procedura

Consultare:

- [“Impostazione dell'esempio di canale di messaggi per AIX, Linux, and Windows” a pagina 154](#) per i dettagli sulla configurazione dei canali di messaggi

- “Esecuzione ed espansione dell'esempio per AIX, Linux, and Windows” a pagina 155 per suggerimenti su come utilizzare altri prodotti, ad esempio CICS, e su come collegare più applicazioni ed uscite utente.

ALW Impostazione dell'esempio di canale di messaggi per AIX, Linux, and Windows

Queste definizioni di oggetto consentono alle applicazioni connesse al gestore code QM1 di inviare messaggi di richieste a una coda denominata PAYROLL su QM2 e di ricevere risposte su una coda denominata PAYROLL.REPLY su QM1 e consente alle applicazioni connesse al gestore code QM2 di richiamare i messaggi di richiesta da una coda locale denominata PAYROLL e di inserire le risposte a tali messaggi di richiesta in una coda denominata PAYROLL.REPLY sul gestore code QM1.

Informazioni su questa attività

Tutte le definizioni di oggetto sono state fornite con gli attributi DESCR e REPLACE. Gli altri attributi forniti sono il minimo richiesto per far funzionare gli esempi. Gli attributi non forniti assumono i valori predefiniti per i gestori code QM1 e QM2.

Non è necessario fornire una definizione della coda remota per consentire la restituzione delle risposte a QM1. Il descrittore del messaggio richiamato dalla coda locale PAYROLL contiene sia il nome della coda di risposta che il nome del gestore code di risposta. Pertanto, fino a quando QM2 può risolvere il nome del gestore code di risposta in quello di una coda di trasmissione sul gestore code QM2, il messaggio di risposta può essere inviato. In questo esempio, il nome del gestore code di risposta è QM1 e quindi il gestore code QM2 richiede una coda di trasmissione con lo stesso nome.

Procedura

- Eseguire i seguenti comandi sul gestore code QM1:

- a) Impostare la definizione della coda remota:

```
DEFINE QREMOTE(PAYROLL.QUERY) DESCR('Remote queue for QM2') REPLACE +
PUT(ENABLED) XMITQ(QM2) RNAME(PAYROLL) RQMNAME(QM2)
```

Nota: La definizione della coda remota non è una coda fisica, ma un mezzo per dirigere i messaggi alla coda di trasmissione, QM2, in modo che possano essere inviati al gestore code QM2.

- b) Impostare la definizione della coda di trasmissione:

```
DEFINE QLOCAL(QM2) DESCR('Transmission queue to QM2') REPLACE +
USAGE(XMITQ) PUT(ENABLED) GET(ENABLED) TRIGGER TRIGTYPE(FIRST) +
INITQ(SYSTEM.CHANNEL.INITQ) PROCESS(QM1.TO.QM2.PROCESS)
```

Quando il primo messaggio viene inserito su questa coda di trasmissione, viene inviato un messaggio trigger alla coda di iniziazione, SYSTEM.CHANNEL.INITQ. L'iniziatore del canale riceve il messaggio dalla coda di iniziazione e avvia il canale identificato nel processo denominato.

- c) Configurare la definizione del canale mittente:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP) +
REPLACE DESCR('Sender channel to QM2') XMITQ(QM2) +
CONNAME('192.0.2.1(1412)')
```

- d) Impostare la definizione del canale ricevente:

```
DEFINE CHANNEL(QM2.TO.QM1) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QM2')
```

- e) Impostare la definizione reply - to_queue:

```
DEFINE QLOCAL(PAYROLL.REPLY) REPLACE PUT(ENABLED) GET(ENABLED) +
DESCR('Reply queue for replies to query messages sent to QM2')
```

La coda di risposta è definita come PUT (ENABLED). Ciò garantisce che i messaggi di risposta possano essere inseriti nella coda. Se le risposte non possono essere inserite nella coda di risposta, vengono inviate alla coda di messaggi non recapitabili su QM1 oppure, se questa coda non è disponibile, rimangono nella coda di trasmissione QM1 sul gestore code QM2. La coda è stata definita come GET (ENABLED) per consentire il richiamo dei messaggi di risposta.

- Eseguire questi comandi sul gestore code QM2: :
 - a) Impostare la definizione della coda locale:

```
DEFINE QLOCAL(PAYROLL) REPLACE PUT(ENABLED) GET(ENABLED) +
DESCR('Local queue for QM1 payroll details')
```

Questa coda è definita come PUT (ENABLED) e GET (ENABLED) per lo stesso motivo della definizione della coda di risposta sul gestore code QM1.

- b) Impostare la definizione della coda di trasmissione:

```
DEFINE QLOCAL(QM1) DESCR('Transmission queue to QM1') REPLACE +
USAGE(XMITQ) PUT(ENABLED) GET(ENABLED) TRIGGER TRIGTYPE(FIRST) +
INITQ(SYSTEM.CHANNEL.INITQ) PROCESS(QM2.TO.QM1.PROCESS)
```

Quando il primo messaggio viene inserito su questa coda di trasmissione, viene inviato un messaggio trigger alla coda di iniziazione, SYSTEM.CHANNEL.INITQ. L'iniziatore del canale riceve il messaggio dalla coda di iniziazione e avvia il canale identificato nel processo denominato.

- c) Configurare la definizione del canale mittente:

```
DEFINE CHANNEL(QM2.TO.QM1) CHLTYPE(SDR) TRPTYPE(TCP) +
REPLACE DESCR('Sender channel to QM1') XMITQ(QM1) +
CONNAME('192.0.2.0(1411)')
```

- d) Impostare la definizione del canale ricevente:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QM1')
```

ALW Esecuzione ed espansione dell'esempio per AIX, Linux, and Windows

Informazioni sull'avvio dell'iniziatore di canali e del listener e suggerimenti per l'espansione su questo scenario.

Informazioni su questa attività

Una volta create queste definizioni, è necessario:

- Avviare l'iniziatore di canali su ciascun gestore code.
- Avviare il listener per ciascun gestore code.

È anche possibile espandere l'esempio.

Procedura

1. Avviare il listener e l'iniziatore del canale.

Consultare [Impostazione della comunicazione per Windows](#) e [Impostazione della comunicazione sui sistemi AIX and Linux](#).

2. È possibile espandere questo esempio:

- L'utilizzo delle comunicazioni LU 6.2 per l'interconnessione con i sistemi CICS e l'elaborazione delle transazioni.
- Aggiunta di ulteriori definizioni di coda, processo e canale per permettere ad altre applicazioni di inviare messaggi tra i due gestori code.
- Aggiunta di programmi di uscita utente sui canali per consentire la codifica dei link, il controllo della sicurezza o l'elaborazione di ulteriori messaggi.
- Utilizzare gli alias del gestore code e gli alias della coda di risposta per comprendere meglio come questi possono essere utilizzati nell'organizzazione della rete del gestore code.

IBM i

Esempio: pianificazione di un canale di messaggi in IBM i



Un esempio dettagliato di come connettere due gestori code IBM i in modo che i messaggi possano essere inviati tra loro. L'esempio riguarda un'applicazione di query payroll connessa al gestore code QM1 che invia messaggi di query payroll a un'applicazione di elaborazione payroll in esecuzione sul gestore code QM2. L'applicazione di query del libro paga ha bisogno delle risposte alle sue query inviate a QM1.

Informazioni su questa attività

L'esempio illustra le preparazioni necessarie per consentire a un'applicazione che utilizza il gestore code QM1 di inserire i messaggi su una coda nel gestore code QM2. Un'applicazione in esecuzione su QM2 può richiamare questi messaggi e inviare risposte a una coda di risposte su QM1.

L'esempio illustra l'utilizzo delle connessioni TCP/IP. L'esempio presuppone che i canali debbano essere attivati per essere avviati quando arriva il primo messaggio sulla coda di trasmissione che stanno gestendo.

Questo esempio utilizza SYSTEM.CHANNEL.INITQ come coda di iniziazione. Questa coda è già stata definita da IBM MQ. È possibile utilizzare una coda di iniziazione differente, ma è necessario definirla personalmente, avviare una nuova istanza dell'iniziatore di canali utilizzando il comando STRMQMCHLI e fornire il nome della propria coda di inizializzazione. Per ulteriori informazioni sull'attivazione dei canali, consultare [Trigget dei canali](#).

Nota:   Un canale di messaggi che utilizza TCP/IP può essere puntato a un IBM Aspera faspio Gateway, che fornisce un tunnel TCP/IP veloce che può aumentare significativamente la velocità di trasmissione di rete. Consultare [Definizione di una connessione Aspera gateway su Linux o Windows](#).

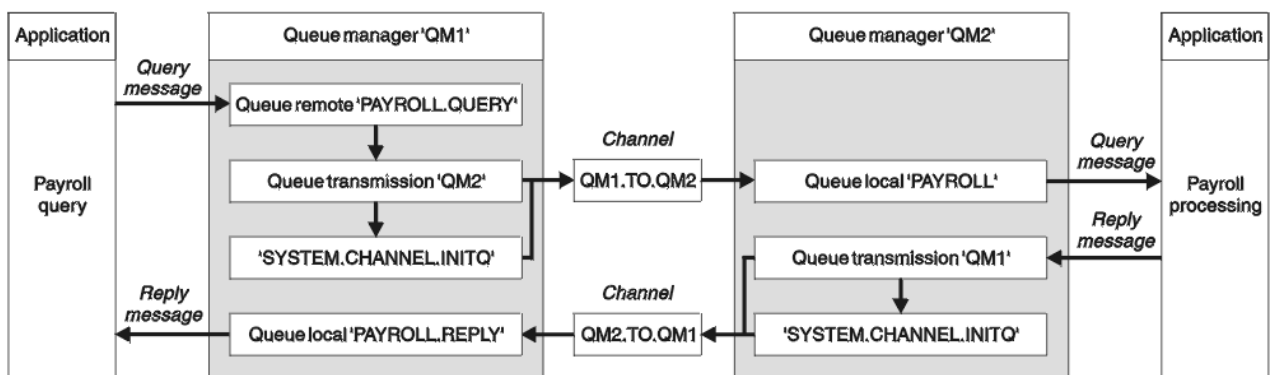


Figura 8. L'esempio di canale dei messaggi per IBM MQ for IBM i

I messaggi di query payroll vengono inviati da QM1 a QM2 su un canale mittente - destinatario denominato QM1.TO.QM2 e i messaggi di risposta vengono inviati di nuovo da QM2 a QM1 su un altro canale mittente - destinatario denominato QM2.TO.QM1. Entrambi questi canali vengono attivati per essere avviati non appena hanno un messaggio da inviare all'altro gestore code.

L'applicazione di query payroll inserisce un messaggio di query nella coda remota "PAYROLL.QUERY" definito in QM1. Questa definizione di coda remota viene risolta nella coda locale "PAYROLL" su QM2. Inoltre, l'applicazione della query payroll specifica che la risposta alla query viene inviata alla coda locale "PAYROLL.REPLY" su QM1. L'applicazione di elaborazione del libro paga richiama i messaggi dalla coda locale "PAYROLL" su QM2 e invia le risposte ovunque siano richieste; in questo caso, la coda locale "PAYROLL.REPLY" su QM1.

Si presuppone che entrambi i gestori code siano in esecuzione su IBM i. Nelle definizioni di esempio, QM1 ha un indirizzo host 192.0.2.0 ed è in ascolto sulla porta 1411. QM2 ha un indirizzo host 192.0.2.1 ed è in ascolto sulla porta 1412. Nell'esempio si assume che questi gestori code siano già definiti sul proprio sistema IBM i e siano disponibili per l'utilizzo.

Le definizioni di oggetto che devono essere create su QM1 sono:

- Definizione coda remota, PAYROLL.QUERY
- Definizione della coda di trasmissione, QM2 (valore predefinito = nome gestore code remoto)
- Definizione del canale mittente, QM1.TO.QM2
- Definizione canale ricevente, QM2.TO.QM1
- Definizione coda di risposta, PAYROLL.REPLY

Le definizioni di oggetto che devono essere create su QM2 sono:

- Definizione coda locale, PAYROLL
- Definizione coda di trasmissione, QM1 (valore predefinito=nome gestore code remoto)
- Definizione canale mittente, QM2.TO.QM1
- Definizione di canale destinatario, QM1.TO.QM2

I dettagli di collegamento vengono forniti nell'attributo CONNAME delle definizioni del canale mittente.

È possibile visualizzare un diagramma della disposizione in [Figura 8 a pagina 156](#).

Procedura

Consultare:

- [“Impostazione dell'agent del canale dei messaggi su IBM i” a pagina 157](#) per i dettagli sulla configurazione dei canali di messaggi
- [“Esecuzione ed espansione dell'esempio per IBM i” a pagina 160](#) per suggerimenti su come collegare più applicazioni e uscite utente.

IBM i

Impostazione dell'agent del canale dei messaggi su IBM i

Le seguenti definizioni di oggetto consentono alle applicazioni connesse al gestore code QM1 di inviare messaggi di richiesta a una coda denominata PAYROLL su QM2, per ricevere risposte su una coda denominata PAYROLL.REPLY su QM1, consente alle applicazioni connesse al gestore code QM2 di richiamare i messaggi di richiesta da una coda locale denominata PAYROLL e di inserire le risposte a tali messaggi di richiesta in una coda denominata PAYROLL.REPLY sul gestore code QM1.

Informazioni su questa attività

Tutte le definizioni di oggetto sono state fornite con gli attributi TEXT. Gli altri attributi forniti sono il minimo richiesto per far funzionare l'esempio. Gli attributi non forniti assumono i valori predefiniti per i gestori code QM1 e QM2.

Non è necessario fornire una definizione della coda remota per consentire la restituzione delle risposte a QM1. Il descrittore del messaggio richiamato dalla coda locale PAYROLL contiene sia il nome della coda di risposta che il nome del gestore code di risposta. Pertanto, fino a quando QM2 può risolvere il nome del gestore code di risposta in quello di una coda di trasmissione sul gestore code QM2, il messaggio di

risposta può essere inviato. In questo esempio, il nome del gestore code di risposta è QM1 e quindi il gestore code QM2 richiede una coda di trasmissione con lo stesso nome.

Procedura

- Eseguire i seguenti comandi sul gestore code QM1:
 - a) Impostare la definizione della coda remota utilizzando il comando CRTMQMQ con i seguenti attributi:

QNAME	'PAYROLL.QUERY'
QTYPE	*RMT
TESTO	'Coda remota per QM2'
PUTENBL	*YES
TMQNOME	'QM2' (valore predefinito = nome gestore code remoto)
NOME RMTQ	'RETRIBUZIONE'
RMTMQMNAME	'QM2'

Nota: La definizione della coda remota non è una coda fisica, ma un mezzo per dirigere i messaggi alla coda di trasmissione, QM2, in modo che possano essere inviati al gestore code QM2.

- b) Configurare la definizione della coda di trasmissione utilizzando il comando CRTMQMQ con i seguenti attributi:

QNAME	QM2
QTYPE	*LCL
TESTO	'Coda di trasmissione a QM2'
USAGE	*TMQ
PUTENBL	*YES
GETENBL	*YES
TRGENBL	*YES
TIPOTRG	*FIRST
NOMEINITQ	SYSTEM.CHANNEL.INITQ
TRIGDATA	QM1.TO.QM2

Quando il primo messaggio viene inserito su questa coda di trasmissione, viene inviato un messaggio trigger alla coda di iniziazione, SYSTEM.CHANNEL.INITQ. L'iniziatore del canale riceve il messaggio dalla coda di iniziazione e avvia il canale identificato nel processo denominato.

- c) Impostare la definizione del canale mittente utilizzando il comando CRTMQCHL con i seguenti attributi:

NOME UTENTE	QM1.TO.QM2
CHLTYPE	*SDR
TRPTYPE	*TCP
TESTO	'Canale mittente a QM2'
TMQNOME	QM2
CONNAME	'192.0.2.1(1412)'

- d) Impostare la definizione del canale destinatario utilizzando il comando CRTMQCHL con i seguenti attributi:

NOME UTENTE	QM2.TO.QM1
CHLTYPE	*RCVR
TRPTYPE	*TCP
TESTO	'Canale ricevente da QM2'

- e) Impostare la definizione della coda reply - to utilizzando il comando CRTMQMQ con i seguenti attributi:

QNAME	PAYROLL.REPLY
QTYPE	*LCL
TESTO	'Coda di risposte per le risposte ai messaggi di query inviati a QM2'
PUTENBL	*YES
GETENBL	*YES

La coda di risposta è definita come PUT (ENABLED). Questa definizione assicura che i messaggi di risposta possano essere inseriti nella coda. Se le risposte non possono essere inserite nella coda di risposta, vengono inviate alla coda di messaggi non recapitabili su QM1 oppure, se questa coda non è disponibile, rimangono nella coda di trasmissione QM1 sul gestore code QM2. La coda è stata definita come GET (ENABLED) per consentire il richiamo dei messaggi di risposta.

- Eseguire questi comandi sul gestore code QM2: :
 - a) Impostare la definizione della coda locale utilizzando il comando CRTMQMQ con i seguenti attributi:

QNAME	payroll
QTYPE	*LCL
TESTO	'Coda locale per dettagli sul libro paga QM1 '
PUTENBL	*YES
GETENBL	*YES

Questa coda è definita come PUT (ENABLED) e GET (ENABLED) per lo stesso motivo della definizione della coda di risposta sul gestore code QM1.

- b) Configurare la definizione della coda di trasmissione utilizzando il comando CRTMQMQ con i seguenti attributi:

QNAME	QM1
QTYPE	*LCL
TESTO	'Coda di trasmissione a QM1'
USAGE	*TMQ
PUTENBL	*YES
GETENBL	*YES
TRGENBL	*YES
TIPOTRG	*FIRST
NOMEINITQ	SYSTEM.CHANNEL.INITQ
TRIGDATA	QM2.TO.QM1

Quando il primo messaggio viene inserito su questa coda di trasmissione, viene inviato un messaggio trigger alla coda di iniziazione, SYSTEM.CHANNEL.INITQ. L'iniziatore del canale riceve il messaggio dalla coda di avvio e avvia il canale identificato nei dati del trigger.

c) Impostare la definizione di canale mittente utilizzando il comando CRTMQMCHL con i seguenti attributi:

NOME UTENTE	QM2.TO.QM1
CHLTYPE	*SDR
TRPTYPE	*TCP
TESTO	'Canale mittente a QM1'
TMQNOME	QM1
CONNNAME	'192.0.2.0(1411)'

d) Impostare la definizione del canale ricevente utilizzando il comando CRTMQMCHL con i seguenti attributi:

NOME UTENTE	QM1.TO.QM2
CHLTYPE	*RCVR
TRPTYPE	*TCP
TESTO	'Canale destinatario da QM1'

IBM i Esecuzione ed espansione dell'esempio per IBM i

Informazioni sull'avvio dell'inziatore di canali e del listener e suggerimenti per l'espansione su questo scenario.

Informazioni su questa attività

Una volta create queste definizioni, è necessario:

- Avviare l'inziatore di canali su ciascun gestore code.
- Avviare il listener per ciascun gestore code.

Le applicazioni possono quindi inviare messaggi tra loro. I canali vengono attivati per essere avviati dal primo messaggio che arriva su ciascuna coda di trasmissione, quindi non è necessario emettere il comando STRMQMCHL.

È anche possibile espandere l'esempio.


Procedura

1. Avviare il listener e l'inziatore del canale.

Consultare [Monitoraggio e controllo dei canali su IBM i](#) per i dettagli sull'avvio di un inziatore di canali e di un listener.

2. È possibile espandere questo esempio:

- Aggiunta di ulteriori definizioni di code e canali per consentire ad altre applicazioni di inviare messaggi tra i due gestori code.
- Aggiunta di programmi di uscita utente sui canali per consentire la codifica dei collegamenti, il controllo della sicurezza o un'ulteriore elaborazione dei messaggi.
- Utilizzare gli alias del gestore code e gli alias della coda di risposta per comprendere meglio il modo in cui questi oggetti possono essere utilizzati nell'organizzazione della rete di gestori code.



 Per una versione di questo esempio che utilizza i comandi MQSC, consultare [“Example: planning a message channel on z/OS”](#) a pagina 161.

How to connect z/OS or MVS queue managers together so that messages can be sent between them. This example involves a payroll query application connected to queue manager QM1 that sends payroll query messages to a payroll processing application running on queue manager QM2. The payroll query application needs the replies to its queries sent back to QM1.

About this task

The example illustrates the preparations needed to allow an application using queue manager QM1 to put messages on a queue at queue manager QM2. An application running on QM2 can retrieve these messages, and send responses to a reply queue on QM1.

The example illustrates the use of both TCP/IP and LU 6.2 connections. The example assumes that channels are to be triggered to start when the first message arrives on the transmission queue they are servicing.

Note:   Un canale di messaggi che utilizza TCP/IP può essere puntato a un IBM Aspera faspio Gateway, che fornisce un tunnel TCP/IP veloce che può aumentare significativamente la velocità di trasmissione di rete. Consultare [Definizione di una connessione Aspera gateway su Linux o Windows](#).

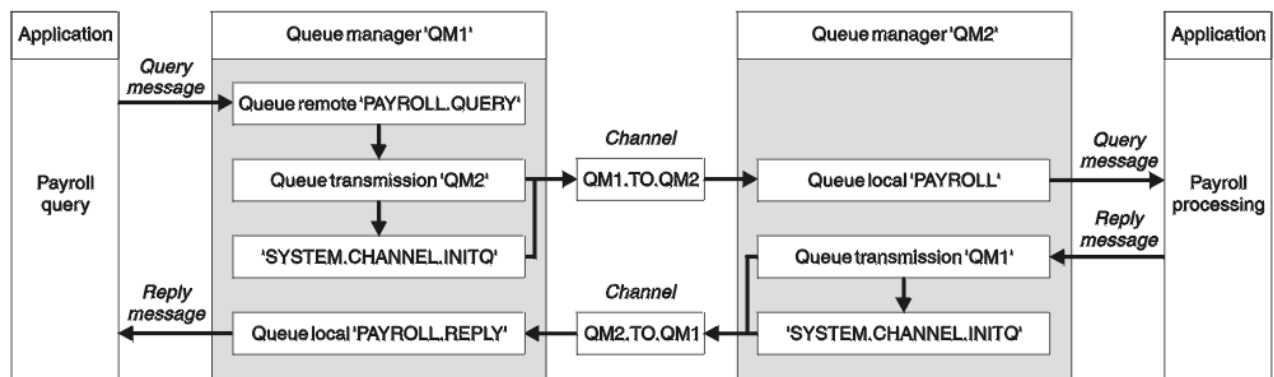


Figure 9. The first example for IBM MQ for z/OS

The payroll query messages are sent from QM1 to QM2 on a sender-receiver channel called QM1.TO.QM2, and the reply messages are sent back from QM2 to QM1 on another sender-receiver channel called QM2.TO.QM1. Both of these channels are triggered to start as soon as they have a message to send to the other queue manager.

The payroll query application puts a query message to the remote queue "PAYROLL.QUERY" defined on QM1. This remote queue definition resolves to the local queue "PAYROLL" on QM2. In addition, the payroll query application specifies that the reply to the query is sent to the local queue "PAYROLL.REPLY" on QM1. The payroll processing application gets messages from the local queue "PAYROLL" on QM2, and sends the replies to wherever they are required; in this case, local queue "PAYROLL.REPLY" on QM1.

Both queue managers are assumed to be running on z/OS. In the example definitions for TCP/IP, QM1 has a host address of 192.0.2.0 and is listening on port 1411, and QM2 has a host address of 192.0.2.1 and is listening on port 1412. In the definitions for LU 6.2, QM1 is listening on a symbolic luname called LUNAME1 and QM2 is listening on a symbolic luname called LUNAME2. The example assumes that these lunames are already defined on your z/OS system and available for use. To define them, see ["Example: setting up IBM MQ cross-platform communication on z/OS"](#) on page 45.

The object definitions that need to be created on QM1 are:

- Remote queue definition, PAYROLL.QUERY
- Transmission queue definition, QM2 (default=remote queue manager name)
- Sender channel definition, QM1.TO.QM2

- Receiver channel definition, QM2.TO.QM1
- Reply-to queue definition, PAYROLL.REPLY

The object definitions that need to be created on QM2 are:

- Local queue definition, PAYROLL
- Transmission queue definition, QM1 (default=remote queue manager name)
- Sender channel definition, QM2.TO.QM1
- Receiver channel definition, QM1.TO.QM2

The example assumes that all the SYSTEM.COMMAND.* and SYSTEM.CHANNEL.* queues required to run DQM have been defined as shown in the supplied sample definitions, **CSQ4INSG** and **CSQ4INSX**.

The connection details are supplied in the CONNAME attribute of the sender channel definitions.

You can see a diagram of the arrangement in [Figure 9 on page 161](#).

Procedure

See:

- [“Setting up the message channel agent on z/OS” on page 162](#) for details on setting up the message channels
- [“Running and expanding the example for z/OS” on page 164](#) for suggestions on how you can connect more applications and user exits.

z/OS

Setting up the message channel agent on z/OS

The following object definitions allow applications connected to queue manager QM1 to send request messages to a queue called PAYROLL on QM2 and also allows applications to receive replies on a queue called PAYROLL.REPLY on QM1. The definitions also allow applications connected to queue manager QM2 to retrieve request messages from a local queue called PAYROLL, and to put replies to these request messages to a queue called PAYROLL.REPLY on queue manager QM1.

About this task

All the object definitions have been provided with the DESCR and REPLACE attributes and are the minimum required to make the example work. The attributes that are not supplied take the default values for queue managers QM1 and QM2.

You do not need to provide a remote queue definition to enable the replies to be returned to QM1. The message descriptor of the message retrieved from local queue PAYROLL contains both the reply-to queue and the reply-to queue manager names. Therefore, as long as QM2 can resolve the reply-to queue manager name to that of a transmission queue on queue manager QM2, the reply message can be sent. In this example, the reply-to queue manager name is QM1 and so queue manager QM2 requires a transmission queue of the same name.

Procedure

- Run the following commands on queue manager QM1:
 - a) Setup the remote queue definition:

```
DEFINE QREMOTE(PAYROLL.QUERY) DESCR('Remote queue for QM2') REPLACE +
PUT(ENABLED) XMITQ(QM2) RNAME(PAYROLL) RQMNAME(QM2)
```

Note: The remote queue definition is not a physical queue, but a means of directing messages to the transmission queue, QM2, so that they can be sent to queue manager QM2.

- b) Setup the transmission queue definition:

```
DEFINE QLOCAL(QM2) DESCR('Transmission queue to QM2') REPLACE +
USAGE(XMITQ) PUT(ENABLED) GET(ENABLED) TRIGGER TRIGTYPE(FIRST) +
TRIGDATA(QM1.TO.QM2) INITQ(SYSTEM.CHANNEL.INITQ)
```

When the first message is put on this transmission queue, a trigger message is sent to the initiation queue, SYSTEM.CHANNEL.INITQ. The channel initiator gets the message from the initiation queue and starts the channel identified in the trigger data. The channel initiator can only get trigger messages from the SYSTEM.CHANNEL.INITQ queue, so do not use any other queue as the initiation queue.

c) Setup the sender channel definition:

For a TCP/IP connection:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP) +
REPLACE DESCR('Sender channel to QM2') XMITQ(QM2) +
CONNAME('192.0.2.1(1412)')
```

For an LU 6.2 connection:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(LU62) +
REPLACE DESCR('Sender channel to QM2') XMITQ(QM2) +
CONNAME('LUNAME2')
```

d) Setup the receiver channel definition:

For a TCP/IP connection:

```
DEFINE CHANNEL(QM2.TO.QM1) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QM2')
```

For an LU 6.2 connection:

```
DEFINE CHANNEL(QM2.TO.QM1) CHLTYPE(RCVR) TRPTYPE(LU62) +
REPLACE DESCR('Receiver channel from QM2')
```

e) Setup the reply-to queue definition:

```
DEFINE QLOCAL(PAYROLL.REPLY) REPLACE PUT(ENABLED) GET(ENABLED) +
DESCR('Reply queue for replies to query messages sent to QM2')
```

The reply-to queue is defined as PUT(ENABLED) which ensures that reply messages can be put to the queue. If the replies cannot be put to the reply-to queue, they are sent to the dead-letter queue on QM1 or, if this queue is not available, remain on transmission queue QM1 on queue manager QM2. The queue has been defined as GET(ENABLED) to allow the reply messages to be retrieved.

- Run the following commands on queue manager QM2:

a) Setup the local queue definition:

```
DEFINE QLOCAL(PAYROLL) REPLACE PUT(ENABLED) GET(ENABLED) +
DESCR('Local queue for QM1 payroll details')
```

This queue is defined as PUT(ENABLED) and GET(ENABLED) for the same reason as the reply-to queue definition on queue manager QM1.

b) Setup the transmission queue definition:

```
DEFINE QLOCAL(QM1) DESCR('Transmission queue to QM1') REPLACE +
USAGE(XMITQ) PUT(ENABLED) GET(ENABLED) TRIGGER TRIGTYPE(FIRST) +
TRIGDATA(QM2.TO.QM1) INITQ(SYSTEM.CHANNEL.INITQ)
```

When the first message is put on this transmission queue, a trigger message is sent to the initiation queue, SYSTEM.CHANNEL.INITQ. The channel initiator gets the message from the initiation queue and starts the channel identified in the trigger data. The channel initiator can only get trigger messages from SYSTEM.CHANNEL.INITQ so do not use any other queue as the initiation queue.

c) Setup the sender channel definition:

For a TCP/IP connection:

```
DEFINE CHANNEL(QM2.TO.QM1) CHLTYPE(SDR) TRPTYPE(TCP) +  
REPLACE DESCR('Sender channel to QM1') XMITQ(QM1) +  
CONNAME('192.0.2.0(1411)')
```

For an LU 6.2 connection:

```
DEFINE CHANNEL(QM2.TO.QM1) CHLTYPE(SDR) TRPTYPE(LU62) +  
REPLACE DESCR('Sender channel to QM1') XMITQ(QM1) +  
CONNAME('LUNAME1')
```

d) Setup the receiver channel definition:

For a TCP/IP connection:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP) +  
REPLACE DESCR('Receiver channel from QM1')
```

For an LU 6.2 connection:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(LU62) +  
REPLACE DESCR('Receiver channel from QM1')
```

z/OS

Running and expanding the example for z/OS

Information about starting the channel initiator and listener and suggestions for expanding on this example.

About this task

Once these definitions have been created, you need to:

- Start the channel initiator on each queue manager.
- Start the listener for each queue manager.

The applications can then send messages to each other. Because the channels are triggered to start by the arrival of the first message on each transmission queue, you do not need to issue the START CHANNEL MQSC command.

You can also expand the example.

Procedure

1. Start the channel initiator and listener.

See [Starting a channel initiator](#), and [Starting a channel listener](#) for details on how to start a channel initiator and listener.

2. You can expand this example by:

- Adding more queues, and channel definitions to allow other applications to send messages between the two queue managers.
- Adding user exit programs on the channels to allow for link encryption, security checking, or additional message processing.

- Using queue manager aliases and reply-to queue aliases to understand more about how these aliases can be used in the organization of your queue manager network.

Example: planning a message channel for z/OS using queue sharing groups

This example illustrates the preparations needed to allow an application using queue manager QM3 to put a message on a queue in a queue sharing group that has queue members QM4 and QM5, and also shows the IBM MQ commands (MQSC) that you can use in IBM MQ for z/OS for distributed queuing with queue sharing groups.

About this task

Ensure you are familiar with the example in [“Example: planning a message channel on z/OS” on page 161](#) before trying this one. This example expands the payroll query scenario of that example, to show how to add higher availability of query processing by adding more serving applications to serve a shared queue.

The payroll query application is now connected to queue manager QM3 and puts a query to the remote queue 'PAYROLL QUERY' defined on QM3. This remote queue definition resolves to the shared queue 'PAYROLL' hosted by the queue managers in the queue sharing group QSG1. The payroll processing application now has two instances running, one connected to QM4 and one connected to QM5.

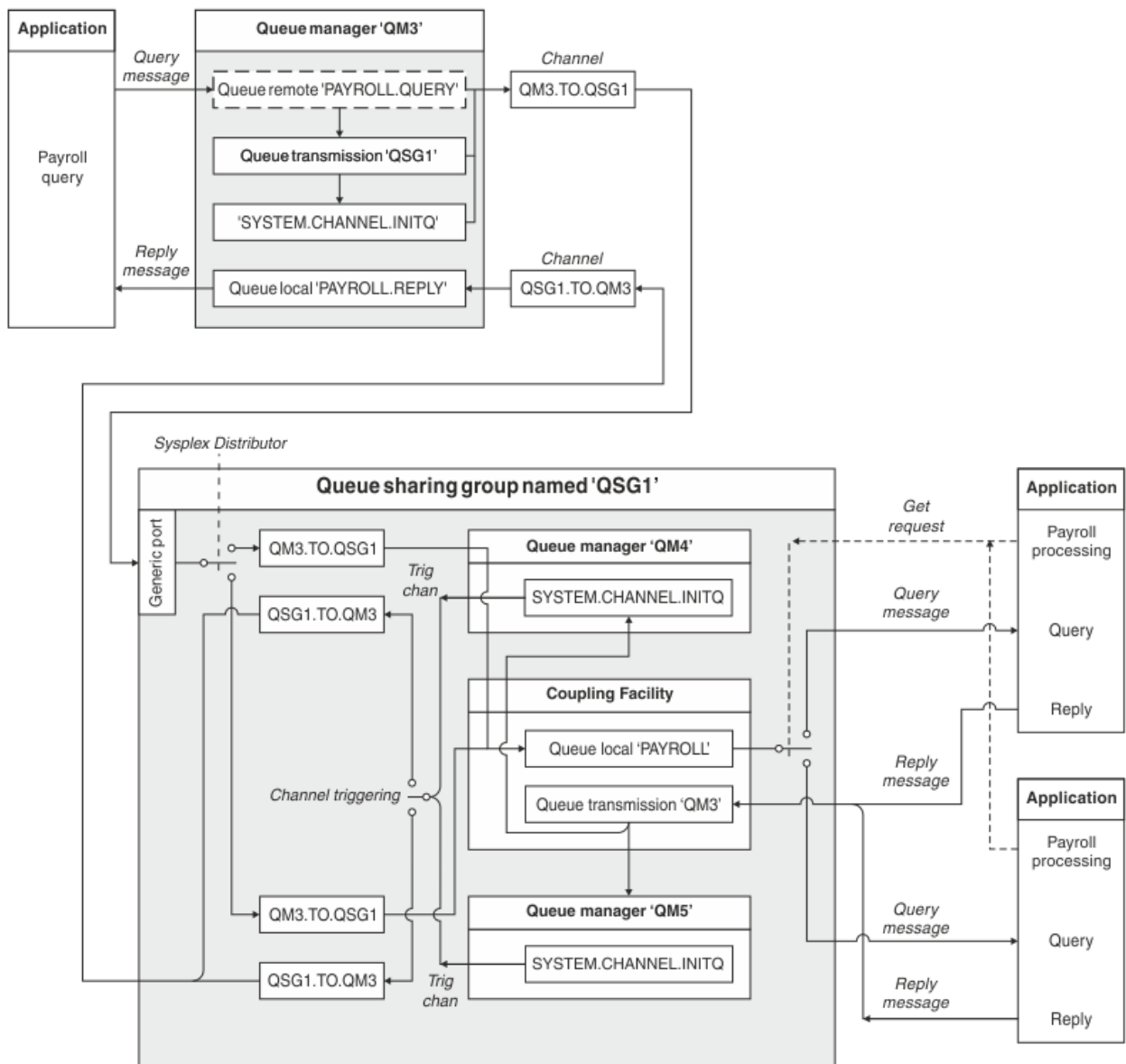


Figure 10. Message channel planning example for IBM MQ for z/OS using queue sharing groups

All three queue managers are assumed to be running on z/OS. In the example definitions for TCP/IP, QM4 has a VIPA address of MVSIP01 and QM5 has a VIPA address of MVSIP02. Both queue managers are listening on port 1414. The generic address that Sysplex Distributor provides for this group is QSG1.MVSIP. QM3 has a host address of 192.0.2.0 and is listening on port 1411.

In the example definitions for LU6.2, QM3 is listening on a symbolic luname called LUNAME1. The name of the generic resource defined for VTAM for the lunames listened on by QM4 and QM5 is LUQSG1. The example assumes that they are already defined on your z/OS system and are available for use. To define them see [“Establishing an LU 6.2 connection into a queue sharing group” on page 50.](#)

In this example QSG1 is the name of a queue sharing group, and queue managers QM4 and QM5 are the names of members of the group.

Procedure

See:

- [“Setting up the queue sharing group definitions and a queue manager QM3 not in the queue sharing group” on page 167](#) for details on setting up the definitions.

- “Running the queue sharing group example for z/OS” on page 168 for details on starting the channel initiators and listeners for each queue manager.

z/OS Setting up the queue sharing group definitions and a queue manager QM3 not in the queue sharing group

Producing the following object definitions for one member of the queue sharing group makes them available to all the other members. QM3 is not a member of the queue sharing group.

About this task

Queue managers QM4 and QM5 are members of the queue sharing group. The definitions produced for QM4 are also available for QM5.

The coupling facility list structure is assumed to be called 'APPLICATION1'. If it is not called 'APPLICATION1', you must use your own coupling facility list structure name for the example.

As QM3 is not a member of the queue sharing group you need the object definitions for that queue manager to allow it to put messages to a queue in the queue sharing group.

Procedure

- Setup the shared objects for the queue sharing group definition:
 - a) Use the following commands to setup the shared object definitions that are stored in Db2, and their associated messages that are stored within the coupling facility.

```
DEFINE QLOCAL(PAYROLL) QSGDISP(SHARED) REPLACE PUT(ENABLED) GET(ENABLED) +
CFSTRUCT(APPLICATION1) +
DESCR('Shared queue for payroll details')

DEFINE QLOCAL(QM3) QSGDISP(SHARED) REPLACE USAGE(XMITQ) PUT(ENABLED) +
CFSTRUCT(APPLICATION1) +
DESCR('Transmission queue to QM3') TRIGGER TRIGTYPE(FIRST) +
TRIGDATA(QSG1.TO.QM3) GET(ENABLED) INITQ(SYSTEM.CHANNEL.INITQ)
```

- Use the following commands to setup the group object definitions that are stored in Db2®. Each queue manager in the queue sharing group creates a local copy of the defined object.

- a) Setup the sender channel:

Sender channel definition for a TCP/IP connection:

```
DEFINE CHANNEL(QSG1.TO.QM3) CHLTYPE(SDR) QSGDISP(GROUP) TRPTYPE(TCP) +
REPLACE DESCR('Sender channel to QM3') XMITQ(QM3) +
CONNNAME('192.0.2.0(1411)')
```

Sender channel definition for an LU 6.2 connection:

```
DEFINE CHANNEL(QSG1.TO.QM3) CHLTYPE(SDR) QSGDISP(GROUP) TRPTYPE(LU62) +
REPLACE DESCR('Sender channel to QM3') XMITQ(QM3) +
CONNNAME('LUNAME1')
```

- b) Setup the receiver channel:

Receiver channel definition for a TCP/IP connection:

```
DEFINE CHANNEL(QM3.TO.QSG1) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QM3') QSGDISP(GROUP)
```

Receiver channel definition for an LU 6.2 connection:

```
DEFINE CHANNEL(QM3.TO.QSG1) CHLTYPE(RCVR) TRPTYPE(LU62) +
REPLACE DESCR('Receiver channel from QM3') QSGDISP(GROUP)
```

- Setup queue manager QM3 object definitions.

a) Setup the CONNAME

The CONNAME for this channel is the generic address of the queue sharing group, which varies according to transport type.

For a TCP/IP connection:

```
DEFINE CHANNEL(QM3.TO.QSG1) CHLTYPE(SDR) TRPTYPE(TCP) +
REPLACE DESCR('Sender channel to QSG1') XMITQ(QSG1) +
CONNAME('QSG1.MVSIP(1414)')
```

For an LU 6.2 connection:

```
DEFINE CHANNEL(QM3.TO.QSG1) CHLTYPE(SDR) TRPTYPE(LU62) +
REPLACE DESCR('Sender channel to QSG1') XMITQ(QSG1) +
CONNAME('LUQSG1') TPNAME('MQSERIES') MODENAME('#INTER')
```

b) Setup the other definitions.

These definitions are required for the same purposes as those used in the sub topics for [“Example: planning a message channel on z/OS”](#) on page 161.

```
DEFINE QREMOTE(PAYROLL.QUERY) DESCR('Remote queue for QSG1') REPLACE +
PUT(ENABLED) XMITQ(QSG1) RNAME(APPL) RQMNAME(QSG1)
```

```
DEFINE QLOCAL(QSG1) DESCR('Transmission queue to QSG1') REPLACE +
USAGE(XMITQ) PUT(ENABLED) GET(ENABLED) TRIGGER TRIGTYPE(FIRST) +
TRIGDATA(QM3.TO.QSG1) INITQ(SYSTEM.CHANNEL.INITQ)
```

```
DEFINE CHANNEL(QSG1.TO.QM3) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QSG1')
```

```
DEFINE CHANNEL(QSG1.TO.QM3) CHLTYPE(RCVR) TRPTYPE(LU62) +
REPLACE DESCR('Receiver channel from QSG1')
```

```
DEFINE QLOCAL(PAYROLL.REPLY) REPLACE PUT(ENABLED) GET(ENABLED) +
DESCR('Reply queue for replies to query messages sent to QSG1')
```

z/OS

Running the queue sharing group example for z/OS

Information about starting the channel initiators and listeners.

About this task

After you have created the required objects, you need to:

- Start the channel initiator for all three queue managers.
- Start the listeners for both queue managers.

Procedure

1. Start the channel initiators.

See [Starting a channel initiator](#) for details on how to start a channel initiator.

2. Start the listeners.

See [Starting a channel listener](#) for details on how to start a listener.

For a TCP/IP connection, each member of the group must have a group listener started that is listening on port 1414.


```
STA LSTR PORT(1414) IPADDR(MVSIP01) INDISP(GROUP)
```

The previous entry starts the listener on QM4, for example.

For an LU6.2 connection, each member of the group must have a group listener started that is listening on a symbolic luname. This luname must correspond to the generic resource LUQSG1.

```
STA LSTR PORT(1411)
```

The previous entry starts the listener on QM3.

Utilizzo di un alias per fare riferimento a una libreria MQ

È possibile definire un alias per fare riferimento a una libreria MQ nel proprio JCL, piuttosto che utilizzare direttamente il nome della libreria MQ. Quindi, se il nome della libreria MQ viene modificato, è necessario solo eliminare e ridefinire l'alias.

Esempio

Il seguente esempio definisce un alias MQM.SCSQANLE per fare riferimento a MQ library MQM.V600.SCSQANLE:

```
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DELETE (MQM.SCSQANLE)
DEFINE ALIAS (NAME(MQM.SCSQANLE) RELATE(MQM.V600.SCSQANLE))
/*
```

Quindi, fare riferimento a MQM.V600.SCSQANLE in JCL, utilizzare l'alias MQM.SCSQANLE.

Nota: I nomi libreria e alias devono trovarsi nello stesso catalogo, quindi utilizzare lo stesso qualificatore di livello superiore per entrambi; in questo esempio, il qualificatore di livello superiore è MQM.

Managed File Transfer Riferimento di configurazione

Informazioni di riferimento che consentono di configurare Managed File Transfer.

L'utilizzo delle variabili di ambiente nelle proprietà MFT

È possibile utilizzare le variabili di ambiente nelle proprietà Managed File Transfer che rappresentano ubicazioni di file o directory. Ciò consente alle ubicazioni dei file o delle directory utilizzate durante l'esecuzione di parti del prodotto di variare in base all'ambiente corrente (ad esempio, l'utente che esegue un comando).

Le seguenti proprietà accettano ubicazioni di file o directory e possono quindi contenere variabili di ambiente:

- agentQMGrAuthenticationCredentialsFile
- agentSslKeyStore
- agentSslKeyStoreCredentialsFile
- agentSslTrustStore
- agentSslTrustStoreCredentialsFile
- cdNodeKeystoreCredentialsFile
- cdNodeTruststoreCredentialsFile
- cdTmpDir
- cdNodeKeystore

- cdNodeTruststore
- commandPath
- connectionQMgrAuthenticationCredentialsFile
- connectionSslKeyStore
- connectionSslKeyStoreCredentialsFile
- connectionSslTrustStore
- connectionSslTrustStoreCredentialsFile
- coordinationSslKeyStore
- coordinationSslKeyStoreCredentialsFile
- coordinationQMgrAuthenticationCredentialsFile
- coordinationSslTrustStore
- coordinationSslTrustStoreCredentialsFile
- exitClassPath
- exitNativeLibraryPath
- javaCoreTriggerFile
- loggerQMgrAuthenticationCredentialsFile
- sandboxRoot
- transferRoot
- wmqfte.database.credentials.file

Esempio per Windows

Windows In questo esempio su un sistema Windows , un utente `fteuser` che utilizza una variabile di ambiente `USERPROFILE`:

```
wmqfte.database.credentials.file=%USERPROFILE%\logger\mqmftcredentials.xml
```

Si risolve nel percorso file seguente:

```
C:\Users\fteuser\logger\mqmftcredentials.xml
```

Esempio per AIX and Linux

Linux **AIX** In questo esempio su un sistema UNIX , un utente `fteuser` che utilizza una variabile di ambiente `HOME`:

```
transferRoot=$HOME/fte/
```

Si risolve nel percorso file seguente:

```
/home/fteuser/fte/
```

Riferimenti correlati

[“File MFT coordination.properties” a pagina 199](#)

Il file `coordination.properties` specifica i dettagli di collegamento al gestore code di coordinamento. Poiché diverse installazioni di Managed File Transfer potrebbero condividere lo stesso gestore code di coordinamento, è possibile utilizzare un link simbolico a un file `coordination.properties` comune su un'unità condivisa.

[“File MFT command.properties” a pagina 203](#)

Il file `command.properties` specifica il gestore code comandi a cui connettersi quando si immettono i comandi e le informazioni richieste da Managed File Transfer per contattare tale gestore code.

“Il file `MFT agent.properties`” a pagina 175

Ogni Managed File Transfer Agent ha il proprio file delle proprietà, `agent.properties`, che deve contenere le informazioni che un agente utilizza per connettersi al proprio gestore code. Il file `agent.properties` può anche contenere proprietà che modificano il comportamento dell'agent.

Proprietà SSL/TLS per MFT

“File `MFT logger.properties`” a pagina 207

Il programma di registrazione Managed File Transfer ha una serie di proprietà di configurazione. Specificare queste propriet ... nel file `logger.properties`, che si trova nella directory `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

MFT Proprietà dell'agent per uscite utente

Formato file delle proprietà bridge di protocollo

Formato file di definizione del processo Connect:Direct

Formato file delle proprietà del nodo Connect:Direct

File `MFT installation.properties`

Il file `installation.properties` specifica il nome della serie predefinita di opzioni di configurazione. Questa voce punta Managed File Transfer ad una serie strutturata di indirizzari e file delle proprietà che contengono la configurazione da utilizzare. Generalmente, il nome di una serie di opzioni di configurazione è il nome del gestore code di coordinamento associato.

Questo file viene creato dal programma di installazione e può essere modificato utilizzando il comando **`fteChangeDefaultConfigurationOptions`**.

Il file `installation.properties` si trova nella directory `MQ_DATA_PATH`. Ad esempio, su Windows, l'ubicazione file predefinita è `MQ_DATA_PATH\mqft\installations\installation_name` e su sistemi AIX and Linux, l'ubicazione file predefinita è `/var/mqm/mqft/installations/installation_name`.

Per Redistributable Managed File Transfer Agent, il percorso dati è impostato quando si esegue il comando **`fteCreateEnvironment`**. Se si esegue il comando e si specifica l'ubicazione scelta con il parametro **`-d`**, il percorso dati viene impostato per questa ubicazione. Se non si specifica l'ubicazione con il comando **`fteCreateEnvironment`**, viene creata una directory `mftdata` nella directory root in cui viene estratto Redistributable Managed File Transfer Agent. Il file `installation.properties` per Redistributable Managed File Transfer Agent si trova nella directory `MQ_DATA_PATH\mqft\installations\MFTZipInstall`.

Il file `installation.properties` contiene i seguenti valori:




Tabella 64. Proprietà di base

Nome proprietà	Descrizione	Valore predefinito
<p>Priorità commandMessage</p>	<p>Imposta la priorità dei messaggi interni e dei messaggi di comando per i comandi fteStopAgent, fteCancelTransfer e ftePingAgent .</p> <p>Se si inoltra un numero elevato di richieste di trasferimento per trasferire molti file di piccole dimensioni in rapida successione, ad esempio, le nuove richieste di trasferimento possono essere accodate nella coda comandi dell'agente di origine. I messaggi interni ed esterni hanno la priorità di messaggio IBM MQ predefinita, in modo che i messaggi interni siano bloccati dalle nuove richieste di trasferimento. Ciò può causare il superamento del tempo di negoziazione del trasferimento e il recupero dei trasferimenti.</p> <p>È anche possibile utilizzare la proprietà Priorità commandMessage per impostare la priorità dei messaggi interni di riconoscimento e di riconoscimento - previsti.</p> <p>Per assegnare la priorità ai messaggi Managed File Transfer interni al di sopra delle nuove richieste di trasferimento, impostare questa proprietà su un valore compreso tra 1 (il più basso) e 9 (il più alto).</p> <p>Il valore predefinito della proprietà Priorità commandMessage è 8. Ciò significa che, se l'attributo IBM MQ DEFPRTY (priorità predefinita) su una coda comandi dell'agent è minore o uguale a 7, ai messaggi di negoziazione interni viene assegnata una priorità prima delle nuove richieste di trasferimento. Se il valore dell'attributo DEFPRTY è impostato su 8 o 9, per mantenere l'efficacia della proprietà commandMessagePriority, è necessario modificare la proprietà DEFPRTY o commandMessagePriority.</p>	<p>Il valore predefinito è 8.</p>

Tabella 64. Proprietà di base (Continua)

Nome proprietà	Descrizione	Valore predefinito
commonCredentialsKeyFile	<p>Il nome percorso completo del file contenente la chiave di credenziali utilizzata durante la codifica delle credenziali. Il nome più comune del file delle credenziali MFT è <code>MQMFTCcredentials.xml</code>.</p> <p>Per ulteriori informazioni sull'utilizzo della proprietà <code>KeyFile commonCredentials</code>, consultare Decrypting credentials.</p>	Il percorso completo del file di chiavi
defaultProperties	<p>Il nome della serie predefinita di opzioni di configurazione. Questo valore è il nome di una directory ubicata nella directory di configurazione, che contiene directory e file delle proprietà che specificano le informazioni di configurazione.</p>	Nessun valore predefinito
enableFunctionalFixPack	<p>Il livello di funzione del fix pack da abilitare. Per impostazione predefinita, qualsiasi nuova funzione inclusa con un fix pack non è abilitata. Impostare questa proprietà su un identificativo versione per abilitare le nuove funzioni disponibili con tale versione.</p> <p>È possibile specificare l'identificativo della versione con o senza i caratteri punto (.). Ad esempio, per utilizzare la funzione disponibile con IBM MQ 8.0.0 Fix Pack 2, impostare questa proprietà su <code>8002</code> o <code>8.0.0.2</code>.</p>	Nessun valore predefinito

Tabella 64. Proprietà di base (Continua)

Nome proprietà	Descrizione	Valore predefinito
Formato messagePublication	<p>Consente di specificare il formato di pubblicazione dei messaggi utilizzato dagli agent MFT per i relativi messaggi XML di stato. Questa proprietà può essere impostata sui seguenti valori:</p> <p>messagePublicationFormat=misto I messaggi vengono pubblicati senza MQMD FORMAT (MQFMT_NONE), ad eccezione dei messaggi pubblicati nella struttura ad albero degli argomenti /LOG, che vengono pubblicati nel formato MQMD di MQFMT_STRING.</p> <p>messagePublicationFormat=MQFMT_NONE I messaggi vengono pubblicati senza MQMD FORMAT.</p> <p>messagePublicationFormat=MQFMT_STRING I messaggi vengono pubblicati in formato stringa.</p>	messagePublicationFormat=misto
<p> z/OS-specifico:</p>		
productID	<p>Tipo di prodotto rispetto al quale deve essere registrato l'utilizzo MFT:</p> <ul style="list-style-type: none"> • Prodotto Managed File Transfer autonomo. (MFT è productID). • Parte di un prodotto IBM MQ Advanced . (ADVANCED è il productID). •  Parte di un prodotto IBM MQ Advanced for z/OS Value Unit Edition . (ADVANCEDVUE è productID). <p>Consultare Informazioni sul prodotto di report per ulteriori informazioni sulla registrazione dell'utilizzo del prodotto.</p> <p> Questa proprietà viene ignorata su Multiplatforme.</p>	MFT

Il seguente testo è un esempio del contenuto di un file installation.properties .

```
defaultProperties=ERIS
```

ERIS è il nome di una directory che si trova nella stessa directory del file `installation.properties`. La directory ERIS contiene directory e file delle proprietà che descrivono una serie di opzioni di configurazione.

Concetti correlati

[Opzioni di configurazione MFT su Multiplatforms](#)

Riferimenti correlati

[Opzioni `fteChangeDefaultConfiguration`](#)

Il file `MFT agent.properties`

Ogni Managed File Transfer Agent ha il proprio file delle proprietà, `agent.properties`, che deve contenere le informazioni che un agente utilizza per connettersi al proprio gestore code. Il file `agent.properties` può anche contenere proprietà che modificano il comportamento dell'agent.

Il file `agent.properties` viene creato dal programma di installazione o dal comando **`fteCreateAgent`**, **`fteCreateBridgeAgent`** o **`fteCreateCDAgent`**. È possibile utilizzare uno qualsiasi di questi comandi con l'indicatore **`-f`** per modificare le proprietà del gestore code dell'agent di base e le proprietà dell'agent avanzate associate al tipo di agent che si sta creando. Per modificare o aggiungere proprietà avanzate dell'agent, è necessario modificare il file in un editor di testo.

Multi Su Multiplatforms, il file `agent.properties` per un agente si trova nella directory `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name`.

z/OS Su z/OS, il file `agent.properties` è `$BFG_CONFIG variable/mqft/config/coordination_qmgr_name/agents/agent_name`.

Se si modifica il file `agent.properties`, è necessario riavviare l'agent per rendere effettive le modifiche.

È possibile utilizzare le variabili di ambiente in alcune proprietà Managed File Transfer che rappresentano le ubicazioni di file o directory. Ciò consente di utilizzare le ubicazioni dei file o delle directory durante l'esecuzione di parti del prodotto per variare in base alle modifiche dell'ambiente, ad esempio l'utente che sta eseguendo il processo. Per ulteriori informazioni, consultare [“L'utilizzo delle variabili di ambiente nelle proprietà MFT”](#) a pagina 169.

Windows

Nota: Su Windows, due proprietà:

- `windowsService`
- `windowsServiceVersion`

vengono aggiunti nel file `agent.properties` dai comandi MFT utilizzati per impostare l'esecuzione di un agente come un servizio Windows.

Non aggiungere le proprietà o modificarle manualmente, in quanto ciò impedirà il corretto funzionamento dell'agent.

Per ulteriori informazioni sulla configurazione di un agente per l'avvio come aWindows servizio, vedi [Iniziare unMFT agente come aWindows servizio](#).

Proprietà agent di base

Ogni file MFT `agent.properties` contiene le seguenti proprietà dell'agente di base:

Nome proprietà	Descrizione	Valore predefinito
<code>agentName</code>	Il nome dell'agent. Il nome dell'agente deve essere conforme alle convenzioni di denominazione degli oggetti IBM MQ. Per ulteriori informazioni, consultare “Convenzioni di denominazione degli oggetti MFT” a pagina 235.	Nessun valore predefinito

Tabella 65. Proprietà agent di base (Continua)		
Nome proprietà	Descrizione	Valore predefinito
agentDesc	La descrizione dell'agent - se si sceglie di creare una descrizione.	Nessun valore predefinito
agentQMgr	Il nome del gestore code dell'agente.	Nessun valore predefinito
Host agentQMgr	Il nome host o l'indirizzo IP del gestore code dell'agent.	Nessun valore predefinito
Porta agentQMgr	Il numero di porta utilizzato per le connessioni client al gestore code dell'agente.	1414
Canale agentQMgr	Il nome del canale SVRCONN utilizzato per connettersi al gestore code dell'agente.	SYSTEM.DEF.SVRCONN
agentType	Il tipo di agent: <ul style="list-style-type: none"> • Agent non bridge standard (STANDARD) • Agent bridge di protocollo (BRIDGE) • agent bridge Connect:Direct (CD_BRIDGE) • Agent integrato come utilizzato da IBM Integration Bus (EMBEDDED) • Agent integrato Sterling File Gateway (SFG) 	STANDARD

Se non si specifica un valore per la proprietà Host agentQMgr, per impostazione predefinita viene utilizzata la modalità di bind.

Se si specifica un valore per la proprietà Host agentQMgrma non si specificano i valori per la porta agentQMgre le proprietà del canale agentQMgr, un numero di porta 1414 e un canale SYSTEM.DEF.SVRCONN vengono utilizzati per impostazione predefinita.

Proprietà avanzate dell'agent

Managed File Transfer fornisce inoltre proprietà di agent più avanzate che consentono di configurare gli agent. Se si desidera utilizzare una delle seguenti proprietà, modificare manualmente il file `agent.properties` per aggiungere le proprietà avanzate richieste. Le parentesi, le virgole (,) e le barre retroverse (\) sono caratteri speciali nei comandi MFT e devono essere preceduti da un carattere barra retroversa (\).

Windows I percorsi dei file su Windows possono essere specificati utilizzando doppie barre rovesciate (\\) come separatore o utilizzando singole barre (/). Per ulteriori informazioni sull'escape dei caratteri nei file delle proprietà Java, consultare la documentazione di Oracle [Javadoc per la classe Proprietà](#).

- [Proprietà dimensione agent](#)
- [Proprietà codepage](#)
- [Proprietà del comando](#)
- [Proprietà di connessione](#)
- [Connect:Direct proprietà bridge](#)
- [Proprietà di File to message e Message to file agent](#)
- [Proprietà generali dell'agent](#)
- [Proprietà alta disponibilità](#)
- [Proprietà Input / Output](#)
- [Proprietà del log di trasferimento](#)
- [Proprietà del supporto multicanale](#)
- [Proprietà di più istanze](#)
- [Proprietà del controller di processo](#)
- [Proprietà bridge di protocollo](#)
- [Proprietà del log dell'agent bridge di protocollo](#)


- Proprietà delle code
- Proprietà di monitoraggio risorse
- Proprietà della directory root
- Proprietà Scheduler
- Proprietà di sicurezza
- Proprietà SSL/TLS
- Proprietà timeout
- Proprietà timeout di recupero trasferimento
- Proprietà di traccia e registrazione
- Proprietà del limite di trasferimento
- Proprietà della routine di uscita utente
- IBM MQ proprietà di compressione client
-  z/OS
- Altre proprietà

Tabella 66. Proprietà agent avanzate: dimensione agent

Nome proprietà	Descrizione	Valore predefinito
Intervallo agentCheckpoint	<p>L'intervallo in frame completi di dati tra cui viene preso un punto di controllo per scopi di recupero. Questa è una proprietà avanzata e per la maggior parte delle configurazioni Managed File Transfer non è necessario modificarne il valore.</p> <p>Se si verifica un problema che fa sì che il trasferimento venga ripristinato, il trasferimento può essere ripristinato solo su un limite del punto di controllo. Quindi, maggiore è questo valore (con valori di agentChunkSize, agentWindowSize e agentFrameSize), maggiore è il tempo necessario all'agent per ripristinare i trasferimenti. Per le reti Managed File Transfer affidabili in cui i trasferimenti raramente entrano in uno stato di recupero, può essere utile aumentare questo valore per aumentare le prestazioni complessive.</p>	1
Dimensione agentChunk	<p>La dimensione di ogni porzione di trasferimento per il trasferimento dei dati file. Pertanto, indica la dimensione massima dei messaggi IBM MQ trasferiti tra gli agent di origine e di destinazione. Questa è una proprietà avanzata e per la maggior parte delle configurazioni Managed File Transfer non è necessario modificarne il valore.</p> <p>Questo valore viene negoziato tra l'agent di origine e l'agent di destinazione e viene utilizzato il maggiore dei due valori. Se si desidera modificare il valore di questa proprietà, modificare il valore sia sull'agente di origine che sull'agente di destinazione.</p> <p>agentChunkLa dimensione è un valore intero. Ad esempio: agentChunkDimensione = 10240 imposta la dimensione della porzione su 10 KB.</p>	262144 - byte (equivalente a 256 KB)
Dimensione agentFrame	<p>Il numero di finestre per il frame di trasferimento. Questa è una proprietà avanzata e per la maggior parte delle configurazioni Managed File Transfer non è necessario modificarne il valore.</p> <p>Per le reti con latenza elevata, l'aumento di questo valore può migliorare le prestazioni complessive in quanto fa sì che l'agent abbia più blocchi di messaggi attivi contemporaneamente.</p> <p>Il valore di questa proprietà, moltiplicato per la dimensione agentWindow, moltiplicato per la dimensione agentChunk, indica il limite superiore del consumo di memoria dell'agent per ogni trasferimento. Ad esempio, 262144 - byte chunk x 10 x 5 = 12.5 MB per ogni trasferimento.</p> <p>Nota: se la dimensione dei file trasferiti in un singolo trasferimento è inferiore a 12.5 MB aumentando questa proprietà non ha alcun effetto sulle prestazioni del trasferimento.</p>	5

Tabella 66. Proprietà agent avanzate: dimensione agent (Continua)

Nome proprietà	Descrizione	Valore predefinito
Dimensione agentWindow	<p>Il numero di blocchi per ciascuna finestra. Questa è una proprietà avanzata e per la maggior parte delle configurazioni Managed File Transfer non è necessario modificarne il valore.</p> <p>Per le reti con latenza elevata, l'aumento di questo valore può migliorare le prestazioni complessive. Ciò è dovuto al fatto che l'agent ha più blocchi di messaggi attivi contemporaneamente e riduce la frequenza con cui i messaggi di riconoscimento vengono inviati di nuovo all'agent di origine.</p> <p>Il valore di questa proprietà, moltiplicato per la dimensione agentFrame, moltiplicato per la dimensione agentChunk, denota il limite superiore del consumo di memoria dell'agent per ogni trasferimento e indica il limite superiore dei dati del messaggio IBM MQ sulla coda dati dell'agent di destinazione. Ad esempio, 262144 - byte chunk x 10 x 5 = un limite superiore di 12.5 MB, per ogni trasferimento.</p> <p>Nota: se la dimensione dei file trasferiti in un singolo trasferimento è inferiore a 12.5 MB, l'aumento del valore di questa proprietà non ha alcun effetto sulle prestazioni del trasferimento.</p>	10

Tabella 67. Proprietà agent avanzate: pagina Codice

Nome proprietà	Descrizione	Valore predefinito
agentCcsid	La codepage con cui l'agent si connette al gestore code dell'agent. Se si specifica un valore per agentCcsid, è necessario specificare anche un valore per il nome agentCcsid. Per informazioni su come visualizzare le codepage note per JVM, consultare il parametro <code>-hsc</code> nel comando <code>fteCreateBridgeAgent</code> .	1208
Nome agentCcsid	La rappresentazione Java di agentCcsid. Se si specifica un valore per il nome agentCcsid, è necessario specificare anche un valore per agentCcsid.	UTF8

Tabella 68. Proprietà avanzate dell'agent: Comando

Nome proprietà	Descrizione	Valore predefinito
maxCommandHandlerThreads	Controlla il numero di thread disponibili per l'analisi iniziale e l'elaborazione dei messaggi di comando di trasferimento. Quando sono attivi, i thread richiedono una connessione al gestore code ma i thread rilasciano la connessione quando sono inattivi.	5
Output maxCommand	Il numero massimo di byte memorizzati per l'output del comando. Questa proprietà è valida per i comandi specificati per una chiamata gestita e i comandi preSource, postSource, preDestination, postDestination per un trasferimento gestito. Ciò limita la lunghezza dell'output del comando scritto nel log di trasferimento sul sistema SYSTEM.FTE.	10240
Tentativi maxCommand	Il numero massimo di tentativi per un comando consentito dall'agente. Questa proprietà si applica a comandi specificati per una chiamata gestita e ai comandi preSource, postSource, preDestination, postDestination per un trasferimento gestito.	9
Attesa maxCommand	L'attesa massima, in secondi, tra i tentativi consentiti dall'agent. Questa proprietà si applica a comandi specificati per una chiamata gestita e ai comandi preSource, postSource, preDestination, postDestination per un trasferimento gestito.	60
Timeout immediateShutdown	<p>Per un arresto immediato di un agent, è possibile utilizzare questa proprietà per specificare la quantità massima di tempo, in secondi, in cui un agent attende il completamento dei propri trasferimenti prima di forzare un arresto.</p> <p>Nota: Non modificare il valore di questa proprietà in meno del valore predefinito di 10 secondi. Un arresto immediato di un agente richiede tempo sufficiente per terminare eventuali processi esterni. Se il valore di questa proprietà è troppo basso, i processi potrebbero essere lasciati in esecuzione.</p> <p>Se il valore 0 è specificato per questa proprietà, l'agent attende l'arresto di tutti i trasferimenti in sospeso. Se per questa proprietà viene specificato un valore non valido, viene utilizzato il valore predefinito.</p>	10

Tabella 69. Proprietà avanzate dell'agente: Connessione		
Nome proprietà	Descrizione	Valore predefinito
Percorso javaLibrary	Quando ci si connette a un gestore code in modalità di bind, Managed File Transfer deve avere accesso alle librerie di bind IBM MQ Java . Per impostazione predefinita, Managed File Transfer ricerca le librerie di bind nell'ubicazione predefinita definita da IBM MQ. Se le librerie di bind si trovano in un'ubicazione diversa, utilizzare questa proprietà per specificare l'ubicazione delle librerie di bind.	Nessuna

Tabella 70. Proprietà agent avanzate: bridge Connect:Direct		
Nome proprietà	Descrizione	Valore predefinito
cdNode	Proprietà obbligatoria se si desidera utilizzare il bridge Connect:Direct . Il nome del nodo Connect:Direct da utilizzare per trasferire i messaggi dall'agent bridge Connect:Direct ai nodi di destinazione Connect:Direct . Questo nodo fa parte del bridge Connect:Direct , non il nodo remoto che è l'origine o la destinazione del trasferimento. Per ulteriori informazioni, consultare Il bridge Connect:Direct .	Nessun valore predefinito
Host cdNode	Il nome host o l'indirizzo IP del nodo Connect:Direct da utilizzare per il trasferimento dei file dall'agent bridge Connect:Direct ai nodi di destinazione (il nodo bridge Connect:Direct). Nella maggior parte dei casi, il nodo bridge Connect:Direct si trova sullo stesso sistema dell'agent bridge Connect:Direct . In questi casi, il valore predefinito di questa proprietà, che è l'indirizzo IP del sistema locale, è corretto. Se il sistema dispone di più indirizzi IP o se il nodo bridge Connect:Direct si trova su un sistema diverso rispetto all'agent bridge Connect:Direct e i relativi sistemi condividono un file system, utilizzare questa proprietà per specificare il nome host corretto per il nodo bridge Connect:Direct . Se non è stata impostata la proprietà cdNode , questa proprietà viene ignorata.	Il nome host o l'indirizzo IP del sistema locale
Porta cdNode	Il numero di porta del nodo bridge Connect:Direct utilizzato dalle applicazioni client per comunicare con il nodo. Nella documentazione del prodotto Connect:Direct , questa porta è indicata come porta API. Se non è stata impostata la proprietà cdNode , questa proprietà viene ignorata.	1363
cimpDir	L'ubicazione in cui memorizzare temporaneamente i file sul sistema in cui è in esecuzione l'agent bridge Connect:Direct prima che vengano trasferiti al nodo Connect:Direct di destinazione. Questa proprietà specifica il percorso completo della directory in cui i file vengono memorizzati temporaneamente. Ad esempio, se cdTmpDir è impostato su /tmp , i file vengono collocati temporaneamente nella directory /tmp . L'agent bridge Connect:Direct e il nodo bridge Connect:Direct devono essere in grado di accedere alla directory specificata da questo parametro utilizzando lo stesso nome percorso. Considerare questa limitazione quando si pianifica di installare il bridge Connect:Direct. Se possibile, creare l'agent sul sistema in cui è ubicato il nodo Connect:Direct che fa parte del bridge Connect:Direct. Se l'agent e il nodo si trovano su sistemi separati, la directory deve essere su un file system condiviso ed essere accessibile da entrambi i sistemi che utilizzano lo stesso nome percorso. Per ulteriori informazioni sulle configurazioni supportate, consultare Il bridge Connect:Direct . Se non è stata impostata la proprietà cdNode , questa proprietà viene ignorata. Il valore di questa proprietà può contenere variabili di ambiente. Per ulteriori informazioni, consultare " L'utilizzo delle variabili di ambiente nelle proprietà MFT " a pagina 169.	<code>value_of_java.io.tmpdir /cdbridge-agentName</code>
cdTrace	Indica se l'agent esegue la traccia dei dati inviati tra l'agent bridge Connect:Direct e il relativo nodo Connect:Direct . Il valore di questa proprietà può essere true o false.	No
cdMaxConnectionRetries	Il numero massimo di tentativi di connessione Connect:Direct , per un trasferimento file in cui non è stata ancora effettuata una connessione riuscita, prima che il trasferimento abbia esito negativo.	-1 (un numero infinito di tentativi)
cdMaxPartialWorkConnectionRetries	Il numero massimo di tentativi di connessione Connect:Direct , per un trasferimento file in cui un precedente tentativo di connessione è riuscito e il lavoro di trasferimento è stato completato, prima che il trasferimento abbia esito negativo.	-1 (un numero infinito di tentativi)

Tabella 70. Proprietà agent avanzate: bridge Connect:Direct (Continua)


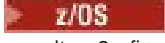


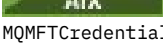

Nome proprietà	Descrizione	Valore predefinito
cdMaxWaitForProcessEndStats	Il tempo massimo, in millisecondi, di attesa per la disponibilità delle informazioni di completamento del processo Connect:Direct all'interno delle informazioni sulle statistiche del nodo Connect:Direct, dopo che il processo è terminato, prima che il trasferimento file venga ritenuto non riuscito. In genere le informazioni sono disponibili immediatamente, ma in determinate condizioni di errore le informazioni non vengono pubblicate. In queste condizioni il trasferimento file non riesce dopo l'attesa per la quantità di tempo specificata da questa proprietà.	60000
Nome cdApp	Il nome dell'applicazione utilizzato dall'agent bridge Connect:Direct per connettersi al nodo Connect:Direct che fa parte del bridge.	Managed File Transfer <i>versione corrente</i> , dove <i>versione corrente</i> è il numero di versione del prodotto.
Intervallo cdNodeLocalPort	La gamma di porte locali da utilizzare per connessioni socket tra l'agent bridge Connect:Direct e il nodo Connect:Direct che fa parte del bridge. Il formato di questo valore è un elenco separato da virgole di valori o intervalli. Per default, il sistema operativo seleziona i numeri di porta locali.	Nessuna
Protocollo cdNode	Il protocollo utilizzato dall'agent bridge Connect:Direct per connettersi al nodo Connect:Direct che fa parte del bridge. I valori validi sono: <ul style="list-style-type: none"> • TCPIP • SSL • TLS 	TCPIP
Keystore cdNode	Il percorso del keystore utilizzato per le comunicazioni protette tra l'agent bridge Connect:Direct e il nodo Connect:Direct che fa parte del bridge. Se non è stata impostata la proprietà del protocollo cdNodesu SSL o TLS, questa proprietà viene ignorata. Il valore di questa proprietà può contenere variabili di ambiente. Per ulteriori informazioni, consultare "L'utilizzo delle variabili di ambiente nelle proprietà MFT" a pagina 169.	Nessuna
cdNodeKeystoreType	Il formato file del keystore specificato dalla proprietà keystore cdNode. I seguenti valori sono validi: jks e pkcs12. Se non è stata impostata la proprietà del protocollo cdNodesu SSL o TLS, questa proprietà viene ignorata.	JKS
File cdNodeKeystoreCredentials	Il percorso per il file che contiene le credenziali del keystore cdNode. Il valore di questa proprietà può contenere variabili di ambiente. Per ulteriori informazioni, consultare "L'utilizzo delle variabili di ambiente nelle proprietà MFT" a pagina 169.  Per i dettagli sulla creazione dei file delle credenziali, consultare MFT e IBM MQ connection authentication  Per i dettagli sulla creazione dei file delle credenziali, consultare Configuring MQMFTCredentials.xml on z/OS .	Il valore predefinito per questa proprietà è:    \$HOME/ MQMFTCredentials.xml  %USERPROFILE%/ MQMFTCredentials.xml
Truststore cdNode	Il percorso del truststore utilizzato per le comunicazioni protette tra l'agent bridge Connect:Direct e il nodo Connect:Direct che fa parte del bridge. Se non è stata impostata la proprietà del protocollo cdNodesu SSL o TLS, questa proprietà viene ignorata. Il valore di questa proprietà può contenere variabili di ambiente. Per ulteriori informazioni, consultare "L'utilizzo delle variabili di ambiente nelle proprietà MFT" a pagina 169.	Nessuna
cdNodeTruststoreType	Il formato file del truststore specificato dalla proprietà del truststore cdNode. I seguenti valori sono validi: jks e pkcs12. Se non è stata impostata la proprietà del protocollo cdNodesu SSL o TLS, questa proprietà viene ignorata.	JKS

Tabella 70. Proprietà agent avanzate: bridge Connect:Direct (Continua)




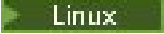


Nome proprietà	Descrizione	Valore predefinito
cdNodeTruststoreCredentials	<p>Il percorso del file che contiene le credenziali del truststore cdNode.</p> <p>Il valore di questa proprietà può contenere variabili di ambiente. Per ulteriori informazioni, consultare “L'utilizzo delle variabili di ambiente nelle proprietà MFT” a pagina 169.</p> <p> Per i dettagli sulla creazione dei file delle credenziali, consultare MFT e IBM MQ connection authentication</p> <p> Per dettagli sulla creazione dei file delle credenziali, consultare Configuring MQMFTCredentials.xml on z/OS.</p>	<p>Il valore predefinito per questa proprietà è:</p> <p> z/OS</p> <p> Linux</p> <p> AIX \$HOME/MQMFTCredentials.xml</p> <p> Windows %USERPROFILE%/MQMFTCredentials.xml</p>
logCDProcess	<p>Il livello di registrazione del processo Connect:Direct registrato nel log eventi dell'agent nel file output0.log. I valori che questa proprietà può avere sono Nessuno o Errori o Tutti.</p>	Nessuna

Tabella 71. Proprietà agent avanzate: da file a messaggio e da messaggio a agent file

Nome proprietà	Descrizione	Valore predefinito
deleteTmpFileAfterRenameFailure	<p>L'impostazione di questa proprietà su un valore false garantisce che i file temporanei non vengano eliminati dalla destinazione se l'operazione di ridenominazione non riesce. In questo caso, i dati trasferiti rimangono nella destinazione in un file temporaneo (.part). È possibile rinominare manualmente questo file in un secondo momento. Per impostazione predefinita, questa proprietà ha il valore true. Questa proprietà si applica sia ai trasferimenti da messaggio a file che da file a file.</p>	vero, true
enableQueueInputOutput	<p>Per impostazione predefinita, l'agent non può leggere i dati da una coda di origine o scrivere i dati in una coda di destinazione come parte di un trasferimento. L'impostazione di questo valore su true consente all'agent di eseguire trasferimenti da file a messaggio e da messaggio a file. Il valore di questa proprietà può essere true o false.</p>	No
Output enableSystemQueueInput	<p>Specifica se l'agente può leggere o scrivere nelle code di sistema IBM MQ. Le code di sistema hanno come prefisso il qualificativo SYSTEM.</p> <p>Nota: Le code di sistema vengono utilizzate da IBM MQ, Managed File Transfer e altre applicazioni per trasmettere informazioni importanti. La modifica di questa proprietà consente all'agent di accedere a queste code. Se si abilita questa proprietà, utilizzare il sandboxing dell'utente per limitare le code a cui l'agent può accedere.</p>	No
Output enableClusterQueueInput	<p>Specifica se l'agente può leggere o scrivere in code con cluster IBM MQ.</p> <p>Nota: È necessario specificare la proprietà dell'agent di output enableClusterQueueInputin aggiunta alla proprietà enableQueueInputOutput.</p>	No
maxDelimiterMatchLength	<p>Il numero massimo di caratteri che possono corrispondere all'espressione regolare Java utilizzata per suddividere un file di testo in più messaggi come parte di un trasferimento file - a - messaggio.</p>	5
maxInputOutputMessageLunghezza	<p>La lunghezza massima, in byte, di un messaggio letto da una coda di origine o scritto in una coda di destinazione da un agent. La proprietà maxInputOutputMessageLength dell'agent di origine in un trasferimento determina quanti byte possono essere letti da un messaggio sulla coda di origine. La proprietà maxInputOutputMessageLength dell'agent di destinazione in un trasferimento determina quanti byte possono essere scritti in un messaggio sulla coda di destinazione. Se la lunghezza del messaggio supera il valore di questa proprietà, il trasferimento ha esito negativo con un errore. Questa proprietà non influisce sulle code interne Managed File Transfer. Per informazioni sulla modifica di questa proprietà, consultare Guida per l'impostazione degli attributi MQ e delle MFT proprietà associate alla dimensione del messaggio.</p>	1048576

Tabella 71. Proprietà agent avanzate: da file a messaggio e da messaggio a agent file (Continua)

Nome proprietà	Descrizione	Valore predefinito
monitorGroupRetryLimit	<p>Il numero massimo di volte in cui un controllo attiva nuovamente un trasferimento da messaggio a file se il gruppo di messaggi è ancora presente nella coda. Il numero di volte in cui i trigger di trasferimento messaggio - file vengono determinati dal conteggio di backout MQMD del primo messaggio del gruppo.</p> <p>Se l'agent viene riavviato, il monitoraggio attiva nuovamente un trasferimento anche se il numero di volte in cui i trigger di trasferimento superano il valore di monitorGroupRetryLimit. Se questo comportamento causa il numero di volte in cui il trigger del trasferimento supera il valore di monitorGroupRetryLimit, l'agent scrive un errore nel relativo log eventi.</p> <p>Se il valore -1 è specificato per questa proprietà, il controllo attiva di nuovo il trasferimento un numero illimitato di volte fino a quando la condizione del trigger non viene soddisfatta.</p>	10

Tabella 72. Proprietà agent avanzate: Generale

Nome proprietà	Descrizione	Valore predefinito
Limite agentStatusPublishRate	<p>La velocità massima, in secondi, con cui l'agent ripubblica il relativo stato a causa di una modifica nello stato del trasferimento file.</p> <p>Se si imposta questa proprietà su un valore troppo piccolo, le prestazioni della rete IBM MQ potrebbero essere influenzate negativamente.</p>	30
agentStatusPublishRateMin	<p>La frequenza minima in secondi con cui l'agent pubblica il proprio stato. Questo valore deve essere maggiore o uguale al valore della proprietà agentStatusPublishRateLimit.</p>	300
enableMemoryAllocationChecking	<p>Il valore di questa proprietà può essere <code>true</code> o <code>false</code>. Determina se Managed File Transfer Agent verifica che sia disponibile memoria sufficiente per eseguire un trasferimento prima che venga accettato. Il controllo viene eseguito sia sull'agente di origine che su quello di destinazione. Se la memoria disponibile non è sufficiente, il trasferimento viene rifiutato.</p> <p>Quando si calcola la memoria richiesta per un trasferimento, viene utilizzata la memoria massima richiesta dal trasferimento. Pertanto, il valore potrebbe essere maggiore della memoria effettiva utilizzata dal trasferimento. Per questo motivo, il numero di trasferimenti simultanei che è possibile eseguire potrebbe essere ridotto se la proprietà enableMemoryAllocationChecking è impostata su <code>true</code>. Si consiglia di impostare la proprietà su <code>true</code> solo se si verificano problemi con Managed File Transfer che non riescono a causa di errori di memoria esaurita. I trasferimenti che probabilmente consumeranno grandi quantità di memoria sono trasferimenti file - to - message e messaggi - to - file dove le dimensioni dei messaggi sono grandi.</p>	No
enableDetailedReplyMessages	<p>Il valore di questa proprietà può essere <code>true</code> o <code>false</code>. L'impostazione di questa proprietà su <code>true</code> consente alle risposte delle richieste di trasferimento gestite di contenere informazioni dettagliate sui file trasferiti. Le informazioni dettagliate e il formato sono gli stessi di quelli pubblicati nel log di trasferimento nei messaggi di avanzamento, ovvero l'elemento <code><transferSet></code>. Per ulteriori informazioni, consultare Formati dei messaggi del log di trasferimento file.</p> <p>Le informazioni di risposta dettagliate vengono incluse solo quando la richiesta di trasferimento gestito specifica che sono richieste informazioni di risposta dettagliate. Per specificare questo requisito, impostare l'attributo detailed dell'elemento <code><reply></code> del messaggio di richiesta XML <code>managedTransfer</code> inviato all'agente di origine. Per ulteriori informazioni, vedere Formato del messaggio di richiesta di trasferimento file.</p> <p>È possibile generare più messaggi di risposta per ogni richiesta di trasferimento. Questo numero è uguale al numero di messaggi di avanzamento del log di trasferimenti per il trasferimento più 1 (dove il primo messaggio di risposta è una semplice risposta ACK). Le informazioni dettagliate sono incluse in tutti i messaggi, ad eccezione dei messaggi di risposta ACK, ma il risultato del trasferimento complessivo è incluso solo nell'ultima risposta dettagliata.</p>	vero, true

Tabella 72. Proprietà agent avanzate: Generale (Continua)

Nome proprietà	Descrizione	Valore predefinito
enableUserMetadataOptions	<p>Il valore di questa proprietà può essere <code>true</code> o <code>false</code>. Determina se è possibile utilizzare chiavi note per i metadati definiti dall'utente nelle nuove richieste di trasferimento per fornire ulteriori opzioni di trasferimento. Queste chiavi note iniziano sempre con il prefisso <code>com.ibm.wmqfte..</code>. Di conseguenza, quando la proprietà enableUserMetadataOptions è impostata su <code>true</code>, le chiavi che utilizzano questo prefisso non sono supportate per l'utilizzo definito dall'utente. Quando la proprietà enableUserMetadataOptions è impostata su <code>true</code>, le chiavi attualmente supportate sono le seguenti:</p> <p>com.ibm.wmqfte.insertRecordLineSeparator</p> <p>Per trasferimenti di testo. Quando questa chiave è impostata su <code>true</code>, specifica che quando si leggono i file orientati ai record, come ad esempio <code>z/OS dataset</code>, i separatori di riga devono essere inseriti tra i record.</p> <p>Quando questa chiave è impostata su <code>false</code>, specifica che durante la lettura dei file orientati ai record, i separatori di riga non devono essere inseriti tra i record.</p> <p>com.ibm.wmqfte.newRecordOnLineSeparator</p> <p>Per trasferimenti di testo. Quando questa chiave è impostata su <code>true</code>, specifica che durante la scrittura nei file orientati ai record, come <code>z/OS dataset</code>, i separatori di riga indicano un nuovo record e non vengono scritti come parte dei dati.</p> <p>Quando questa chiave è impostata su <code>false</code>, specifica che, durante la scrittura in file orientati ai record, i separatori di riga devono essere considerati come qualsiasi altro carattere (ovvero, nessuna interruzione di record).</p> <p>com.ibm.wmqfte.convertLineSeparators</p> <p>Per trasferimenti di testo. Specifica se le sequenze di separatori di riga <code>CRLF</code> e <code>LF</code> vengono convertite nella sequenza di separatori di riga richiesta per la destinazione. Questa conversione attualmente ha effetto solo per i casi seguenti:</p> <ul style="list-style-type: none"> • Se la chiave di metadati definita dall'utente com.ibm.wmqfte.newRecordOnLineSeparator è impostata su <code>false</code> e il trasferimento è su un file orientato ai record. • Se la chiave di metadati definita dall'utente com.ibm.wmqfte.com.ibm.wmqfte.insertRecordLineSeparator è impostata su <code>false</code> e il trasferimento proviene da un file orientato ai record. <p>Vedere anche fteCreateTransfer: avviare un nuovo trasferimento file.</p>	No
failTransferOnFirstErrore	<p>Il valore di questa proprietà può essere <code>true</code> o <code>false</code>. Consente a un agent di essere configurato in modo da non eseguire un trasferimento gestito non appena un elemento di trasferimento all'interno di tale trasferimento gestito non riesce.</p> <p>Per abilitare questa funzione, <code>APAR IT03450</code> deve essere applicato sia per l'agent di origine che per l'agent di destinazione e la proprietà failTransferOnFirstFailure deve essere impostata su <code>true</code> nel file <code>agent.properties</code> dell'agent di origine. L'impostazione della proprietà su <code>true</code> sull'agent di destinazione è facoltativa.</p> <p>Quando la proprietà failTransferOnFirstFailure è impostata su <code>true</code>, l'agent avvia l'elaborazione delle richieste di trasferimento gestite come normale. Tuttavia, non appena un elemento di trasferimento ha esito negativo, il trasferimento gestito viene contrassegnato come non riuscito e non vengono elaborati ulteriori elementi di trasferimento. Gli elementi di trasferimento che sono stati elaborati correttamente prima che il trasferimento gestito non fosse riuscito vengono gestiti nel modo seguente:</p> <ul style="list-style-type: none"> • La disposizione di origine per tali elementi di trasferimento viene rispettata. Ad esempio, se la disposizione di origine per l'elemento trasferimento è stata impostata su <code>delete</code>, il file di origine viene eliminato. • I file di destinazione che sono stati scritti rimangono sul file system di destinazione e non vengono eliminati. <p>Se la proprietà failTransferOnFirstFailure non è impostata su <code>true</code> e un trasferimento file gestito contiene più file e uno di questi file non riesce a trasferire, ad esempio perché il file di destinazione già esiste e la proprietà di sovrascrittura è impostata su <code>error</code>, l'agent di origine continua e tenta di trasferire i file rimanenti nella richiesta.</p>	No

Tabella 72. Proprietà agent avanzate: Generale (Continua)		
Nome proprietà	Descrizione	Valore predefinito
itemsPerProgressMessage	<p>Il numero di file trasferiti prima che un agente pubblichi il successivo messaggio di log di avanzamento. Questa proprietà controlla la frequenza con cui i messaggi del log di avanzamento vengono pubblicati nel gestore code di coordinamento durante un trasferimento.</p> <p>Il valore massimo su cui questa proprietà può essere impostata è 1000.</p> <p>Nota: I messaggi di avanzamento includono informazioni su ogni file trasferito dalla pubblicazione dell'ultimo messaggio di avanzamento. L'aumento di questo valore aumenta la dimensione dei messaggi di avanzamento, il che potrebbe influire sulle prestazioni.</p>	50
maxInlineFileSize	<p>Per i trasferimenti da file a file singolo o da file a messaggio, la dimensione massima del file (in byte) che può essere inclusa automaticamente nel messaggio di richiesta di trasferimento iniziale.</p> <p>È possibile utilizzare questa proprietà per aumentare la velocità dei trasferimenti, ma se si imposta la dimensione del file su un valore troppo grande, le prestazioni potrebbero peggiorare. Una dimensione iniziale consigliata per questa proprietà è 100 KB, ma si consiglia di testare accuratamente valori diversi fino a quando non si trova la dimensione file migliore per il sistema.</p> <p>Questa funzione è disattivata per impostazione predefinita o impostando la proprietà maxInlineFileSize su 0.</p>	0

Tabella 73. Proprietà agent avanzate: Alta disponibilità		
Nome proprietà	Descrizione	Valore predefinito
highlyAvailable	Questa proprietà viene letta durante l'avvio dell'agent e, se impostata sul valore true, l'agent viene avviato in modalità alta disponibilità. Se non si specifica la proprietà o si imposta il valore su false, l'agent viene avviato come un agent non altamente disponibile.	No
Intervallo standbyPoll	<p>Questa proprietà viene utilizzata dall'istanza in standby per tentare di aprire la coda condivisa a intervalli specificati.</p> <p>Da IBM MQ 9.3.0, questa proprietà viene utilizzata anche da tutte le istanze per determinare per quanto tempo un'istanza attende tra i tentativi di riconnessione se si disconnette dal relativo gestore code dell'agent.</p> <p>I tentativi vengono ripetuti fino a quando un'istanza non si riconnette al gestore code dell'agent, non apre la coda SYSTEM.FTE.HA.<agent name> (se è già stata registrata come istanza in standby) o non viene arrestata dal comando fteStopAgent.</p>	5 secondi
standbyStatusDiscardTime	<p>Questa proprietà imposta il periodo di tempo per cui l'istanza attiva attende una pubblicazione dello stato da un'istanza in standby.</p> <p>Se non viene ricevuta alcuna pubblicazione da un'istanza in standby, anche dopo questo periodo di tempo, l'istanza attiva rimuove le informazioni sull'istanza in standby dal relativo elenco di istanze in standby.</p> <p>Il valore predefinito è due volte quello della proprietà standbyStatusPublishInterval, in modo che l'istanza attiva attenda più tempo prima di rimuovere l'istanza standby dal suo elenco.</p>	600 secondi
Scadenza standbyStatus	Questa proprietà imposta la scadenza del messaggio di stato standby inserito nella coda comandi di un agente. Il messaggio scade se l'istanza attiva di un agent non elabora questo messaggio.	30 secondi
standbyStatusPublishInterval	Questa proprietà viene utilizzata per impostare la frequenza con cui l'istanza in standby pubblica il proprio stato.	300 secondi


Tabella 74. Proprietà avanzate dell'agente: Input / Output		
Nome proprietà	Descrizione	Valore predefinito
doNotUseTempOutputFile	<p>Per impostazione predefinita, l'agent scrive in un file temporaneo nella destinazione e ridenomina questo file temporaneo con il nome file richiesto una volta completato il trasferimento file. L'impostazione di questo valore su true fa sì che l'agent scriva direttamente nel file di destinazione finale.</p> <p> Su sistemi z/OS, questo comportamento non si applica ai dataset sequenziali, ma ai membri del dataset PDS.</p> <p>Il valore di questa proprietà per un trasferimento è definita dall'agente di destinazione.</p>	No

Tabella 74. Proprietà avanzate dell'agente: Input / Output (Continua)

Nome proprietà	Descrizione	Valore predefinito
Blocco enableMandatory	<p>Quando si accede ai file normali, Managed File Transfer utilizza un blocco condiviso per la scrittura e un blocco esclusivo per la lettura.</p> <p>Windows Il blocco del file Windows è solo consultivo. Quando questa proprietà è impostata su true, Managed File Transfer applica il blocco dei file. Su Windows significa che se un'altra applicazione ha un file aperto, il monitoraggio di tale file non viene attivato fino a quando il file non viene chiuso. Managed File Transfer i trasferimenti che riguardano tale file non riescono.</p> <p>UNIX Su piattaforme di tipi UNIX, il blocco dei file viene eseguito tra i processi. Per le piattaforme di tipo UNIX, l'impostazione di questa proprietà non ha alcun effetto.</p> <p>Questa proprietà si applica solo ai normali agent Managed File Transfer. Managed File Transfer non supporta il meccanismo di blocco file su un server di file. Questa proprietà quindi non funziona per l'agent bridge di protocollo poiché l'agent bridge di protocollo non blocca un file su un server di file durante il trasferimento di un file.</p> <p>Il valore di questa proprietà può essere true o false.</p>	No
ioIdleThreadTimeout	<p>Tempo in millisecondi per cui un thread di input / output del file system rimane inattivo prima che il thread venga arrestato.</p> <p>z/OS Da IBM MQ 9.3.0, questa proprietà non si applica agli agent in esecuzione su IBM MQ for z/OS.</p>	10000
Profondità ioQueue	Il numero massimo di richieste di immissione / emissione da accodare.	10
ioThreadPoolSize	<p>Numero massimo di thread di immissione / emissione del file system disponibili. Generalmente ogni trasferimento utilizza il proprio thread di input / output del file system, ma se il numero di trasferimenti simultanei supera questo limite, i thread di input / output del file system vengono condivisi tra i trasferimenti.</p> <p>Se pensi di avere regolarmente più trasferimenti simultanei in corso rispetto al valore ioThreadPoolSize, potresti vedere un miglioramento aumentando questo valore, in modo che ogni trasferimento abbia il proprio thread di input / output del file system.</p>	10
textReplacementCharacterSequence	<p>Per il trasferimento in modalità testo, se uno dei byte di dati non può essere convertito dalla codepage di origine alla codepage di destinazione, il comportamento predefinito è che il trasferimento file abbia esito negativo.</p> <p>Impostare questa proprietà per consentire il corretto completamento del trasferimento inserendo il valore del carattere specificato. Questo valore di proprietà è un carattere singolo. In genere, un punto interrogativo (?) viene utilizzato per qualsiasi carattere non associabile. Ad esempio, utilizzare questo formato textReplacementCharacterSequence=? dove il punto interrogativo (?) è il carattere di sostituzione. Non è possibile utilizzare uno spazio come carattere di sostituzione.</p>	Nessuna

Tabella 75. Proprietà agent avanzate: log di trasferimento

Nome proprietà	Descrizione	Valore predefinito
logTransfer	<p>Attivare o disattivare la registrazione del trasferimento.</p> <p>I valori possibili sono:</p> <p>informazioni Abilita le informazioni di log di alto livello di un trasferimento. Questo è il valore predefinito.</p> <p>moderare Abilita le informazioni di log di livello intermedio di un trasferimento.</p> <p>emissione di messaggi completi Abilita le informazioni di log dettagliate di un trasferimento.</p> <p>off Disattiva la registrazione del trasferimento</p>	informazioni
logTransferFileSize	Definisce la dimensione massima di un file di log di trasferimento in megabyte	20
File logTransfer	Definisce il numero massimo di file di trasferimento conservati prima che il file meno recente venga eliminato.	5

Tabella 76. Proprietà avanzate dell'agente: supporto multi - canale		
Nome proprietà	Descrizione	Valore predefinito
agentMultipleChannelsEnabled	<p>L'impostazione di questa proprietà su <code>true</code> consente a un Managed File Transfer Agent di inviare messaggi di dati su più canali IBM MQ. In alcuni scenari, l'impostazione di questa proprietà potrebbe migliorare le prestazioni. Tuttavia, abilitare il supporto multicanale solo se esiste un vantaggio dimostrabile in termini di prestazioni. Solo i messaggi inseriti nel SISTEMA.SYSTEM.FTE.DATA.La <code>codadestinationAgentName</code> viene inviata su più canali. Il comportamento per tutti i messaggi rimane invariato.</p> <p>Quando si imposta questa proprietà su <code>true</code>, è necessario completare anche la procedura di configurazione IBM MQ in uno dei seguenti argomenti per abilitare il supporto multicanale:</p> <ul style="list-style-type: none"> Configurazione di un agent MFT per più canali in un cluster Configurazione di un agent MFT per più canali: senza cluster <p>Inoltre, è necessario completare i passi di configurazione standard di IBM MQ richiesti per un agente Managed File Transfer, descritti in Configurazione di MFT per il primo utilizzo.</p> <p>Il valore di questa proprietà può essere <code>true</code> o <code>false</code>.</p>	No
agentMessageBatchSize	Quando configurato con più canali, un agent di origine invia messaggi di dati per un trasferimento su ciascun canale su base round - robin. Questa proprietà controlla il numero di messaggi inviati contemporaneamente su ciascun canale.	5

Tabella 77. Proprietà avanzate dell'agente: gestore code a più istanze		
Nome proprietà	Descrizione	Valore predefinito
agentQMGrStandby	<p>Il nome host e il numero di porta utilizzati per le connessioni client, in formato IBM MQ CONNAME, per l'istanza in standby di un gestore code agent a più istanze definito da agentQMGr. Ad esempio, <code>host_name(port_number)</code></p> <p>L'agent tenta di collegarsi al gestore code in standby quando rileva un errore di connessione interrotta, ad esempio MQRC 2009. Una volta che l'agent viene connesso al gestore code in standby, l'agent rimane connesso fino a quando il gestore code in standby non diventa non disponibile.</p>	Nessun valore predefinito

Tabella 78. Proprietà agent avanzate: controller di processo		
Nome proprietà	Descrizione	Valore predefinito
agentQMGrRetryInterval	L'intervallo, in secondi, tra i controlli sulla disponibilità del gestore code da parte del controller processi dell'agent.	30
Conteggio maxRestart	Il numero massimo di riavvii che possono verificarsi entro l'intervallo di tempo specificato dal valore della proprietà maxRestartInterval. Quando questo valore viene superato, il controller processi dell'agent arresta il riavvio dell'agent ed esegue invece un'azione basata sul valore della proprietà Delay maxRestart.	4
Intervallo maxRestart	L'intervallo, in secondi, con cui il controller di processi dell'agent misura il riavvio dell'agent. Se il numero di riavvii in questo intervallo supera il valore della proprietà maxRestartCount, il controller processi dell'agent arresta il riavvio dell'agent. Invece, il controller di processi dell'agente effettua un'operazione basata sul valore della proprietà Delay maxRestart.	120
Ritardo maxRestart	Determina il funzionamento del controller di processi dell'agent quando la velocità di riavvio dell'agent supera il valore delle proprietà maxRestartCount e maxRestartInterval. Se si specifica un valore inferiore o uguale a zero, il controller di processi dell'agente viene arrestato. Se si specifica un valore maggiore di zero, si tratta del numero di secondi da attendere prima che le informazioni sulla cronologia dei riavvii contenute nel controller processi dell'agent vengano reimpostate e l'agent venga riavviato.	-1

Tabella 79. Proprietà avanzate dell'agente: bridge di protocollo		
Nome proprietà	Descrizione	Valore predefinito
protocolBridgeCredentialConfiguratio n	Il valore di questa proprietà viene passato come una stringa al metodo initialize () delle classi di uscita specificate dalle classi protocolBridgeCredentialExit.	vuoto
protocolBridgeCredentialExitClasses	Specifica un elenco separato da virgole di classi che implementano una routine uscita utente di credenziali del bridge di protocollo. Per ulteriori informazioni, consultare Associazione credenziali per un server file utilizzando le classi di uscita .	Nessun valore predefinito.

Tabella 79. Proprietà avanzate dell'agente: bridge di protocollo (Continua)		
Nome proprietà	Descrizione	Valore predefinito
protocolBridgeDataTimeout	Il timeout in millisecondi che l'agent bridge di protocollo attende per stabilire una connessione dati a un server FTP o per ricevere dati da un server FTP su una connessione già stabilita. Se si imposta questa proprietà su un valore pari a 0, l'agent bridge di protocollo attende indefinitamente. Se il timeout scade, l'agent bridge di protocollo chiude le connessioni dati esistenti al server FTP e tenta di stabilire una nuova connessione dati prima di riprendere il trasferimento corrente. Se il tentativo di stabilire la nuova connessione dati ha esito negativo, anche il trasferimento corrente ha esito negativo.	0
protocolBridgeLogoutBeforeDisconnessioni	Specifica se l'agent bridge di protocollo scollega l'utente dal server di file prima di chiudere la sessione FTP e disconnettersi. Se si imposta questa proprietà su <code>true</code> , l'agent bridge di protocollo immette un comando FTP QUIT sul server di file.	No
protocolBridgePropertiesConfigurazione	Passato come una delle proprietà del bridge al metodo <code>initialize()</code> delle classi di uscita specificate dalla proprietà <code>protocolBridgeServerPropertiesExitClasses</code> .	Nessun valore predefinito
protocolBridgePropertiesExitClasses	Specifica un elenco separato da virgole di classi che implementano una routine uscita utente di proprietà del server bridge di protocollo. Per ulteriori informazioni, consultare ProtocolBridgePropertiesExit2: Looking up protocol file server properties .	Nessun valore predefinito

Tabella 80. Proprietà avanzate dell'agente: Registrazione dell'agent bridge di protocollo		
Nome proprietà	Descrizione	Valore predefinito
agentLog	Il componente della coppia di valori chiave e l'operazione per abilitare o disabilitare la registrazione dei comandi FTP e delle risposte tra i server di file FTP/SFTP/FTPS e Protocol Bridge Agent. Ad esempio: agentLog=on Attiva la registrazione per tutti i componenti agentLog=off Disattivare la registrazione per tutti i componenti agentLog=ftp=on, sftp=on, ftps=off Attivare la registrazione per FTP e SFTP e disattivare per FTPS	Nessun valore predefinito
agentLogFileSize	Definisce la dimensione massima di un file di cattura in megabyte. Uguale al valore predefinito per la dimensione file predefinita di traccia regolare.	20
File agentLog	Definisce il numero massimo di file di cattura conservati prima che il file meno recente venga eliminato. <ul style="list-style-type: none">Il valore predefinito della proprietà dell'agent agentLogFiles è stato modificato da 10 a 5. Ciò significa che da IBM MQ 9.3.0, se è impostato il valore predefinito, possono essere presenti un massimo di cinque file di log eventi dell'agent bridge di protocollo, a partire da <code>agentevent0.log</code> a <code>agentevent4.log</code>. Tuttavia, è possibile modificare questo valore, se necessario.Se l'agent viene migrato da una versione precedente a IBM MQ 9.3.0, è necessario eliminare manualmente i file <code>agentevent5.log</code> in <code>agentevent9.log</code>, se presenti.Tuttavia, la dimensione di ciascun file di log rimane a 20 MB.	Da IBM MQ 9.3.0, il valore predefinito è 5.
Filtro agentLog	Per impostazione predefinita cattura la comunicazione con tutti i server FTP a cui si sta collegando l'agente. Ad esempio: <ul style="list-style-type: none">Filtra su host / indirizzo IP <pre>host=ftpprod.ibm.com, ftp2.ibm.com host=9.182.*</pre>Filtra in base a metadati <pre>metadata="outbound files to xyz corp"</pre>	*

Tabella 81. Proprietà agent avanzate: Coda		
Nome proprietà	Descrizione	Valore predefinito
Prefisso dynamicQueue	Questa proprietà definisce il prefisso da utilizzare durante la creazione di una coda dinamica temporanea.	WMQFTE.*
Nome modelQueue	Questa proprietà definisce il nome della coda del modulo da utilizzare quando si crea una coda dinamica temporanea.	SYSTEM.DEFAULT.MODEL.QUEUE
publicationMDUser	L'ID utente MQMD da associare ai messaggi inviati per essere pubblicati dal gestore code di coordinamento. Se non si imposta questa proprietà, l'ID utente MQMD viene impostato in base alle regole IBM MQ per l'impostazione degli ID utente MQMD.	Nessun valore predefinito

Tabella 82. Proprietà avanzate dell'agent: monitoraggio delle risorse		
Nome proprietà	Descrizione	Valore predefinito
monitorFilepathPlatformSeparator	Specifica se utilizzare i separatori di percorso specifici della piattaforma all'interno della variabile <code>\$FILEPATH</code> . Il valore <code>true</code> utilizza i separatori di percorso specifici della piattaforma. Un valore <code>false</code> utilizza un separatore di percorso barra (/) di stile UNIX su tutte le piattaforme.	vero, true
Polling monitorMaxResourcesIn	Specifica il numero massimo di risorse monitorate da attivare in ciascun intervallo di polling. Ad esempio, se si specifica un modello di monitoraggio di <code>*.txt</code> , un intervallo di polling di 10 secondi e si imposta la proprietà di polling <code>monitorMaxResourcesIn</code> su 10, la proprietà di polling <code>monitorMaxResourcesIn</code> limita l'agent ad attivarsi su un massimo di 10 corrispondenze per ciascun intervallo di polling. Le risorse corrispondenti oltre il limite di 10 vengono attivate in intervalli di polling successivi. Inoltre, è possibile utilizzare la proprietà <code>monitorMaxResourcesInPoll</code> in combinazione con un parametro <code>-bs</code> corrispondente nel comando <code>fteCreateMonitor</code> , ad esempio, per limitare ciascun intervallo di polling all'attivazione di solo un trasferimento. Un valore inferiore o uguale a zero indica che il numero di risorse del monitor attivate in un intervallo di polling è illimitato.	-1
monitorReportTriggerFail	Specifica se le condizioni di errore, nell'ambiente e nella configurazione, rilevate nel monitoraggio vengono riportate come un messaggio di log al SISTEMA SYSTEM.FTE. Un valore di <code>true</code> registra i messaggi. Il valore <code>false</code> non registra i messaggi.	vero, true
monitorReportTriggerNotsoddisfatto	Specifica se un trigger non soddisfatto invia un messaggio di log a SYSTEM.FTE che contiene i dettagli. Un valore di <code>true</code> registra i messaggi. Il valore <code>false</code> non registra i messaggi.	No
monitorReportTriggerSatisfied	Specifica se un trigger soddisfatto invia un messaggio di log al SISTEMA SYSTEM.FTE che contiene i dettagli. Un valore di <code>true</code> registra i messaggi. Il valore <code>false</code> non registra i messaggi.	No
monitorSilenceOnTriggerErrore	Il numero di errori consecutivi del trigger di monitoraggio risorse prima che gli errori non vengano più notificati.	5
monitorStopOnInternalErrore	Il numero di condizioni FFDC interne consecutive del monitoraggio risorse prima che il monitoraggio ne modifichi lo stato in arresto.	10

Tabella 83. Proprietà agent avanzate: directory root

Nome proprietà	Descrizione	Valore predefinito
commandPath	<p>Specifica la serie di percorsi da cui è possibile richiamare i comandi, utilizzando uno dei seguenti metodi:</p> <ul style="list-style-type: none"> Attività dell'agent Ant fte: call Ant task, fte: filecopy o fte: filemove In un messaggio XML passato a un agente, utilizzando uno degli schemi XML del comando Managed File Transfer Agent supportati (ad esempio, managedCall o managedTransfer). <p>Per informazioni sulla sintassi valida del valore della proprietà <code>commandPath</code>, consultare commandPath MFT property.</p> <p>Importante: Prestare particolare attenzione quando si imposta questa proprietà perché qualsiasi comando in uno dei <code>commandPath</code> specificati può essere effettivamente richiamato da un sistema client remoto che è in grado di inviare comandi all'agent. Per questo motivo, per impostazione predefinita, quando si specifica un <code>commandPath</code>:</p> <ul style="list-style-type: none"> Qualsiasi sandbox dell'agent esistente viene configurato dall'agent quando viene avviato in modo che tutte le directory <code>commandPath</code> vengano aggiunte automaticamente all'elenco di directory che hanno negato l'accesso per un trasferimento. Tutte le sandbox utente esistenti vengono aggiornate quando l'agent viene avviato in modo che tutte le directory <code>commandPath</code> (e le relative sottodirectory) vengano aggiunte come elementi <code><exclude></code> agli elementi <code><read></code> e <code><write></code>. Se l'agente non è configurato per utilizzare una sandbox dell'agente o una sandbox dell'utente, una nuova sandbox dell'agente viene creata all'avvio dell'agente che ha le directory <code>commandPath</code> specificate come directory negate. <p>Il valore di questa proprietà può contenere variabili di ambiente.</p> <p>Per ulteriori informazioni, consultare "L'utilizzo delle variabili di ambiente nelle proprietà MFT" a pagina 169.</p> <p>È possibile impostare la proprietà <code>addCommandPathToSandbox</code> su false per sovrascrivere questo comportamento predefinito.</p> <p>Importante: Tenere presente che questa sovrascrittura consente in modo efficace a un client di trasferire qualsiasi comando al sistema agent e richiamare il comando, pertanto deve essere utilizzato con estrema attenzione.</p>	Nessuno - nessun comando può essere richiamato
Sandbox addCommandPathTo	<p>Specifica se le directory specificate dalla proprietà <code>commandPath</code> (e tutte le relative sottodirectory) devono essere aggiunte a:</p> <ul style="list-style-type: none"> Le directory negate per una sandbox agent esistente. Gli elementi <code><exclude></code> per gli elementi <code><read></code> e <code><write></code> per le sandbox utente che sono state definite. Una nuova sandbox agent, se un agent non è stato configurato con una sandbox agent o con una o più sandbox utente. <p>Per informazioni relative alla sintassi valida del valore della proprietà <code>commandPath</code>, consultare commandPath MFT property.</p>	Vero
additionalWildcardSandboxChecking	<p>Specifica se devono essere effettuati ulteriori controlli sui trasferimenti di caratteri jolly per un agent che è stato configurato con un utente o una sandbox dell'agente per limitare le ubicazioni in cui l'agent può trasferire i file da e verso.</p> <p>Quando questa proprietà è impostata su true, viene abilitato il controllo aggiuntivo. Se una richiesta di trasferimento tenta di leggere un'ubicazione esterna alla sandbox definita per la corrispondenza del file del carattere jolly, il trasferimento ha esito negativo. Se ci sono più trasferimenti all'interno di una richiesta di trasferimento e una di queste richieste ha esito negativo a causa del tentativo di leggere un'ubicazione all'esterno della sandbox, l'intero trasferimento ha esito negativo. Se il controllo ha esito negativo, il motivo dell'errore viene fornito in un messaggio di errore (consultare Controlli aggiuntivi per i trasferimenti di caratteri jolly).</p> <p>Se la proprietà viene omessa o impostata su false, non vengono effettuati ulteriori controlli sui trasferimenti con caratteri jolly.</p>	Nessuna

Tabella 83. Proprietà agent avanzate: directory root (Continua)

Nome proprietà	Descrizione	Valore predefinito
sandboxRoot	<p>Specifica la serie di percorsi root da includere ed escludere quando si utilizza il sandboxing. Per informazioni su questa funzione, consultare Utilizzo delle MFT sandbox agent.</p> <p>Separare i percorsi con un separatore di percorso specifico della piattaforma. Anteporre ai percorsi un carattere punto esclamativo (!) per indicare i percorsi come esclusi dalla sandbox. Questa funzione è utile se si desidera escludere una sottodirectory in un percorso root incluso.</p> <p>La proprietà sandboxRoot non è supportata sugli agenti bridge di protocollo. non è possibile specificare insieme la proprietà sandboxRoot e la proprietà userSandboxes.</p> <p>Il valore di questa proprietà può contenere variabili di ambiente. Per ulteriori informazioni, consultare "L'utilizzo delle variabili di ambiente nelle proprietà MFT" a pagina 169.</p>	Nessuno - nessuna sandbox
transferRoot	<p>La directory root predefinita per i percorsi relativi specificati per l'agente. Il valore di questa proprietà può contenere variabili di ambiente. Per ulteriori informazioni, consultare "L'utilizzo delle variabili di ambiente nelle proprietà MFT" a pagina 169.</p>	La directory home dell'utente che ha avviato il processo agent.
HLQ transferRoot	HLQ (ID utente) predefinito per dataset non completi specificati sull'agent	Il nome utente dell'utente che ha avviato il processo agent.
userSandboxes	<p>Limitare l'area del file system da cui è possibile trasferire i file in base al nome utente MQMD dell'utente che richiede il trasferimento. Per ulteriori informazioni, consultare Utilizzo delle sandbox utente MFT.</p> <p>La proprietà userSandboxes non è supportata sugli agent bridge di protocollo. non è possibile specificare insieme la proprietà sandboxRoot e la proprietà userSandboxes.</p>	No

Tabella 84. Proprietà agent avanzate: proprietà Scheduler

Nome proprietà	Descrizione	Valore predefinito
maxSchedulerRunDelay	<p>L'intervallo massimo, in minuti, che l'agente attende per controllare i trasferimenti pianificati. Specificare un numero intero positivo per abilitare questa proprietà. Per ulteriori informazioni sul motivo per cui si desidera utilizzare questa proprietà, consultare Cosa fare se il trasferimento file pianificato non viene eseguito o viene ritardato.</p> <p>Poiché l'agent potrebbe leggere un comando dalla relativa coda comandi nel momento in cui è prevista l'esecuzione dei trasferimenti pianificati, potrebbe verificarsi un ulteriore ritardo prima dell'avvio dei trasferimenti pianificati. In questo caso, il programma di pianificazione viene eseguito immediatamente dopo il completamento del comando.</p>	-1

Tabella 85. Proprietà agent avanzate: Sicurezza





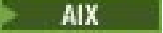

Nome proprietà	Descrizione	Valore predefinito
agentCredentialsKeyFile	Nome del file contenente la chiave della credenziale utilizzata durante la codifica delle credenziali.	Una proprietà stringa che non ha un valore predefinito.
File agentQMGrAuthenticationCredentials	<p>Il nome del percorso del file che contiene le credenziali che devono essere utilizzate durante la connessione al gestore code agent. Il valore di questa proprietà può contenere variabili di ambiente. Per ulteriori informazioni, consultare "L'utilizzo delle variabili di ambiente nelle proprietà MFT" a pagina 169.</p> <p> Per i dettagli sulla creazione dei file delle credenziali, vedere MFT e IBM MQ autenticazione della connessione</p> <p> Per i dettagli sulla creazione del File delle credenziali di autenticazione, consultare Configurazione di MQMFTCredentials.xml su z/OS.</p>	<p>Il valore predefinito per questa proprietà è:</p> <ul style="list-style-type: none">  z/OS  Linux  AIX \$HOME/MQMFTCredentials.xml  Windows %USERPROFILE%/MQMFTCredentials.xml
authorityChecking	<p>Specifica se le funzioni di sicurezza descritte in Limitazione delle autorizzazioni utente sulle azioni dell'agent MFT sono abilitate. L'autorizzazione inquire è un'autorizzazione richiesta su tutte le code di autorizzazione dell'agente.</p>	No

Tabella 85. Proprietà agent avanzate: Sicurezza (Continua)		
Nome proprietà	Descrizione	Valore predefinito
Controlli logAuthority	Il livello di registrazione del controllo di autorizzazione registrato nel log eventi agent nel file output0 . Log . I valori che questa proprietà può avere sono Nessuno o Errori o Tutti.	Nessuna
userIdForClientConnect	L'ID utente che viene passato attraverso le connessioni client a IBM MQ. Se si specifica java , il nome utente riportato dalla JVM viene inoltrato come parte della richiesta di connessione IBM MQ . I valori che questa proprietà può avere sono Nessuno o java.	Nessuna

Tabella 86. Proprietà avanzate dell'agente: SSL/TLS		
Nome proprietà	Descrizione	Valore predefinito
agentSslCipherSpec	<p>Specifica il protocollo, l'algoritmo hash e l'algoritmo di cifratura utilizzati e quanti bit vengono utilizzati nella chiave di cifratura, quando i dati vengono scambiati tra l'agente e il gestore code dell'agente.</p> <p>Il valore di agentSslCipherSpec è un nome CipherSpec . Questo nome CipherSpec è uguale al nome CipherSpec utilizzato sul canale del gestore code dell'agent. Un elenco di nomi CipherSpec è incluso in CipherSpec CipherSpecs e CipherSuites nelle classi IBM MQ per Java e SSL/TLS CipherSpecs e CipherSuites nelle classi IBM MQ per JMS.</p> <p>agentSslCipherSpec è simile a agentSslCipherSuite. Se vengono specificati sia agentSslCipherSuite che agentSslCipherSpec , viene utilizzato il valore di agentSslCipherSpec .</p>	Nessuna
agentSslCipherSuite	<p>Specifica gli aspetti SSL del modo in cui l'agent e il gestore code si scambiano i dati.</p> <p>Il valore di agentSslCipherSuite è un nome CipherSuite . Il nome CipherSuite corrisponde al nome CipherSpec utilizzato sul canale gestore code dell'agent. Per ulteriori informazioni, consultare CipherSuite e le associazioni di nomi CipherSpec.</p> <p>agentSslCipherSuite è simile a agentSslCipherSpec. Se vengono specificati sia agentSslCipherSuite che agentSslCipherSpec , viene utilizzato il valore di agentSslCipherSpec .</p>	Nessuna
agentSslPeerName	Specifica una struttura del DN (distinguished name) che deve corrispondere al nome fornito dal gestore code agent. Il DN (distinguished name) viene utilizzato per controllare il certificato di identificazione presentato dal gestore code sulla connessione.	Nessuna
agentSslTrustStore	<p>Specifica l'ubicazione dei certificati ritenuti attendibili dall'agent. Il valore di agentSslTrustStore è un percorso file. Le parentesi, le virgole (,) e le barre retroverse (\) sono caratteri speciali nei comandi MFT e devono essere preceduti da un carattere barra retroversa (\).</p> <p>Windows I percorsi dei file su Windows possono essere specificati utilizzando doppie barre rovesciate (\\) come separatore o utilizzando singole barre (/).</p> <p>Il valore di questa proprietà può contenere variabili di ambiente.</p>	Nessuna
agentSslKeyStore	<p>Specifica l'ubicazione della chiave privata dell'agent. Il valore di agentSslKeyStore è un percorso file. Le parentesi, le virgole (,) e le barre retroverse (\) sono caratteri speciali nei comandi MFT e devono essere preceduti da un carattere barra retroversa (\).</p> <p>Windows I percorsi dei file su Windows possono essere specificati utilizzando doppie barre rovesciate (\\) come separatore o utilizzando singole barre (/). Questa proprietà è richiesta solo se il gestore code dell'agent richiede l'autenticazione client.</p> <p>Il valore di questa proprietà può contenere variabili di ambiente.</p>	Nessuna
agentSslFipsRequired	Specifica che si desidera abilitare il supporto FIPS a livello di agent. Il valore di questa proprietà può essere true o false. Per ulteriori informazioni, consultare Supporto FIPS in MFT .	No
Tipo agentSslKeyStore	Il tipo di keystore SSL che si desidera utilizzare. Sono supportati i keystore JKS e PKCS#12 . Il valore di questa proprietà può essere jks o pkcs12.	JKS

Tabella 86. Proprietà avanzate dell'agente: SSL/TLS (Continua)




Nome proprietà	Descrizione	Valore predefinito
agentSslKeyStoreCredentialsFile	<p>Il percorso del file che contiene le credenziali per accedere all'archivio chiavi dell'agente.</p> <p>Il valore di questa proprietà può contenere variabili di ambiente. Per ulteriori informazioni, consultare "L'utilizzo delle variabili di ambiente nelle proprietà MFT" a pagina 169.</p> <p> Per i dettagli sulla creazione dei file delle credenziali, vedere MFT e IBM MQ autenticazione della connessione</p> <p> Per i dettagli sulla creazione del File delle credenziali di autenticazione, consultare Configurazione di MQMFTCredentials.xml su z/OS.</p>	<p>Il valore predefinito per questa proprietà è:</p> <p>   \$HOME/ MQMFTCredentials.xml  %USERPROFILE%/MQMFTCredentials.xml</p>
agentSslTrustStore	<p>Il tipo di keystore SSL che si desidera utilizzare. Sono supportati i keystore JKS e PKCS#12 . Il valore di questa proprietà può essere jks o pkcs12.</p>	JKS
agentSslTrustStoreCredentialsFile	<p>Il percorso del file che contiene le credenziali per accedere al truststore dell'agente.</p> <p>Il valore di questa proprietà può contenere variabili di ambiente. Per ulteriori informazioni, consultare "L'utilizzo delle variabili di ambiente nelle proprietà MFT" a pagina 169.</p> <p> Per i dettagli sulla creazione dei file delle credenziali, vedere MFT e IBM MQ autenticazione della connessione</p> <p> Per i dettagli sulla creazione del File delle credenziali di autenticazione, consultare Configurazione di MQMFTCredentials.xml su z/OS.</p>	<p>Il valore predefinito per questa proprietà è:</p> <p>   \$HOME/ MQMFTCredentials.xml  %USERPROFILE%/MQMFTCredentials.xml</p>

Tabella 87. Proprietà agent avanzate: Timeout

Nome proprietà	Descrizione	Valore predefinito
maxTransferNegotiationTime	<p>Il tempo di attesa massimo, in millisecondi, di un trasferimento per il completamento della negoziazione da parte di un agent di destinazione. Se la negoziazione non viene completata entro questo periodo di tempo, il trasferimento viene posto in uno stato di risincronizzazione e consente l'esecuzione di un altro trasferimento, se disponibile.</p> <p>In scenari in cui l'agent di origine o di destinazione è sottoposto a un carico elevato, è possibile che il valore predefinito sia troppo basso per consentire all'agent di rispondere abbastanza rapidamente alla richiesta di negoziazione. Ciò è molto probabile quando un agent di origine ha un numero elevato di monitoraggi delle risorse definiti o quando i relativi monitoraggi delle risorse stanno monitorando directory che contengono un numero elevato di file. Tuttavia, può verificarsi anche quando un numero elevato di richieste di trasferimento viene inoltrato a un agente. Aumentare il valore di questa proprietà a 200.000 o più può essere necessario in tali scenari.</p>	30 000
recoverableTransferRetryInterval	<p>Il tempo di attesa in millisecondi tra il rilevamento di un errore di trasferimento recuperabile e il tentativo di riprendere il trasferimento.</p>	60 000
senderTransferRetryInterval	<p>Il tempo di attesa, in millisecondi, prima che un trasferimento rifiutato venga ritentato perché la destinazione sta già eseguendo il numero massimo di trasferimenti. Il valore minimo è 1000.</p>	30 000
Timeout transferAck	<p>Timeout in millisecondi in cui un trasferimento attende il riconoscimento o i dati dall'altra estremità prima che venga emesso un nuovo tentativo. Questa è una proprietà avanzata e per la maggior parte delle configurazioni Managed File Transfer non è necessario modificarne il valore.</p> <p>I riconoscimenti vengono inviati dall'agent ricevente all'agent mittente ogni volta che viene ricevuta una finestra completa di dati. Per le reti con larghezza di banda limitata o non affidabili e le impostazioni di dimensione agentWindowe agentChunkdi grandi dimensioni, è possibile che il valore predefinito non sia sufficiente. Ciò può causare un trasferimento di dati non necessario tra gli agenti. Pertanto, l'aumento di questo valore potrebbe essere utile e potrebbe ridurre la probabilità che un trasferimento vada in modalità di ripristino a causa di una rete lenta.</p>	60 000
transferAckTimeoutRetries	<p>Numero massimo di tentativi di riconoscimento per un trasferimento senza una risposta prima che l'agent cessi e sposti il trasferimento in uno stato di ripristino</p>	5

Tabella 87. Proprietà agent avanzate: Timeout (Continua)

Nome proprietà	Descrizione	Valore predefinito
xmlConfigReloadInterval	<p>L'intervallo in secondi tra il ricaricamento dei file di configurazione XML da parte dell'agente durante il runtime. Per impedire all'agente di ricaricare i file di configurazione XML durante il runtime, impostare questa proprietà su -1. I seguenti file di configurazione XML sono interessati da questa proprietà:</p> <ul style="list-style-type: none"> • ConnectDirectCredentials.xml • ConnectDirectNodeProperties.xml • ConnectDirectProcessDefinitions.xml • ProtocolBridgeCredentials.xml • ProtocolBridgeProperties.xml • UserSandboxes.xml 	30

Tabella 88. Proprietà agent avanzate: Traccia e registrazione

Nome proprietà	Descrizione	Valore predefinito
javaCoreTriggerFile	<p>Il percorso completo di un'ubicazione file monitorata dall'agent. Se il file esiste nell'ubicazione specificata, l'avvio dell'agent attiverà un Javacore. Dopo aver avviato l'agent, se si aggiorna un file in questa posizione, l'agent attiva di nuovo un file Javacore.</p> <p>Un thread separato esegue il polling di questo file ogni 30 secondi per verificare se il file è stato creato o aggiornato. Se il file è stato creato o aggiornato dall'ultimo polling, l'agent genera un file Javacore nella seguente directory: <code>MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name</code></p> <p>Quando si specifica questa proprietà, l'agent emette il seguente messaggio all'avvio:</p> <pre>BFGAG0092I The <insert_0> file will be used to request JVM diagnostic information.</pre> <p>Il valore di questa proprietà può contenere variabili di ambiente.</p> <p>Per ulteriori informazioni, consultare "L'utilizzo delle variabili di ambiente nelle proprietà MFT" a pagina 169.</p>	Nessuna
traccia	<p>La specifica di traccia da utilizzare quando l'agent viene avviato. Questo è un elenco separato da virgole di classi e / o pacchetti, il carattere uguale e un livello di traccia.</p> <p>Ad esempio, per tracciare la classe <code>com.ibm.wmqfte.agent.Agent</code> e le classi nel package <code>com.ibm.wmqfte.commandhandler</code> dall'avvio dell'agent, aggiungere la seguente voce al file <code>agent.properties</code>:</p> <pre>trace=com.ibm.wmqfte.agent.Agent,com.ibm.wmqfte.commandhandler=all</pre> <p>È possibile specificare più specifiche di traccia in un elenco separato da due punti. Ad esempio:</p> <pre>trace=com.ibm.wmqfte.agent.Agent=all:com.ibm.wmqfte.commandhandler=moderate</pre> <p>La specifica di traccia speciale <code>=all</code> è utilizzata per tracciare l'agent e la JMQUI (Java Message Queuing Interface) che gestisce tutte le comunicazioni con il gestore code dell'agent. Per abilitarla, aggiungere la seguente voce al file <code>agent.properties</code>:</p> <pre>trace==all</pre> <p>Se non diversamente specificato dal rappresentante del supporto IBM, utilizzare la specifica di traccia <code>com.ibm.wmqfte=all</code> come segue:</p> <pre>trace=com.ibm.wmqfte=all</pre>	Nessuna
File outputLog	Il numero totale di <code>output.log</code> file da conservare. Questo valore si applica a un controller di processi dell'agent e all'agent stesso.	5
Dimensione outputLog	La dimensione massima in MB di ogni file <code>output.log</code> prima che l'output venga riportato sul file successivo. Questo valore si applica a un controller di processi dell'agent e all'agent stesso.	1

Tabella 88. Proprietà agent avanzate: Traccia e registrazione (Continua)


Nome proprietà	Descrizione	Valore predefinito
Codifica outputLog	La codifica dei caratteri che l'agente utilizza quando scrive nel file output.log.	La codifica di caratteri predefinita della piattaforma su cui è in esecuzione l'agent.
traceFiles	Il numero totale di file di traccia da mantenere. Questo valore si applica anche al controller processi di un agent e all'agent stesso.	5
traceSize	La dimensione massima, in MB, di ciascun file di traccia prima del ritorno a capo della traccia sul file successivo. Questo valore si applica a un controller di processi dell'agent e all'agent stesso.	20
traceMaxByte	Il limite alla quantità di dati del messaggio emessi nel file di traccia.	4096 byte
Ripristino logTransfer	Quando questa proprietà è impostata su un valore true, ogni volta che un trasferimento immette eventi di diagnostica di ripristino, vengono notificati al log eventi dell'agent nel file output0.log.	Il valore predefinito è true.
logCapture	Cattura i messaggi di richieste di trasferimento inoltrati a questo agent e i messaggi di log pubblicati dall'agent nel gestore code di coordinamento. Questi messaggi acquisiti possono essere utili quando si esegue il debug dei problemi di trasferimento. I messaggi acquisiti vengono memorizzati in file nella directory di log dell'agent denominata capture?.log. ? è un valore numerico. Il file che contiene il numero 0 contiene i messaggi acquisiti più recenti.	No
logCaptureFileSize	Definisce la dimensione massima di un file di cattura in megabyte.	10
File logCapture	Definisce il numero massimo di file di cattura conservati prima che il file meno recente venga eliminato.	10
Filtro logCapture	Un'espressione regolare Java utilizzata dall'agente per corrispondere al nome dell'argomento del messaggio. Vengono catturati solo i messaggi che corrispondono all'espressione regolare.	.* (corrisponde a tutti)
Log resourceMonitor	<p>Coppia valore chiave del monitoraggio risorse e operazione per attivare o disattivare la registrazione.</p> <p>I valori possibili sono:</p> <ul style="list-style-type: none"> • informazioni • moderare • emissione di messaggi completi • off <p>Ad esempio:</p> <ul style="list-style-type: none"> • resourceMonitorLog=MON1,MON2=info:MON3=off Attivare la registrazione per MON1 e MON2e disattivare la registrazione per MON3. • resourceMonitorLog=info Attivare la registrazione a livello info per tutti i monitoraggi risorse. <p>I log di monitoraggio delle risorse vengono scritti in un file denominato resmoneventN.log, dove N indica un numero; ad esempio, resmonevent0.log.</p> <p> Attenzione: Tutti i monitoraggi risorse di un agent scrivono nello stesso file di log.</p> <p>Per ulteriori informazioni, consultare Registrazione dei monitoraggi delle risorse MFT.</p>	informazioni
resourceMonitorLogFile	Definisce la dimensione massima di un file di cattura in megabyte.	20
resourceMonitorLogFiles	<p>Definisce il numero massimo di file di cattura conservati prima che il file meno recente venga eliminato.</p> <ul style="list-style-type: none"> • Il valore predefinito della proprietà dell'agent resourceMonitorLogFiles è stato modificato da 10 a 5. Ciò significa che, da IBM MQ 9.3.0, se è impostato il valore predefinito, possono essere presenti al massimo cinque file di log degli eventi di monitoraggio delle risorse, a partire da resmonevent0.log a resmonevent4.log. Tuttavia, è possibile modificare questo valore, se necessario. • Se l'agent viene migrato da una versione precedente a IBM MQ 9.3.0, è necessario eliminare manualmente i file resmonevent5.log in resmonevent9.log, se presenti. • Tuttavia, la dimensione di ciascun file di log rimane a 20 MB. 	Da IBM MQ 9.3.0, il valore predefinito è 5.

Tabella 89. Proprietà agent avanzate: limite di trasferimento

Nome proprietà	Descrizione	Valore predefinito
Trasferimenti maxDestination	<p>Il numero massimo di trasferimenti simultanei che l'agent di destinazione elabora in qualsiasi momento. Ogni richiesta di trasferimento inoltrata a un agente viene contata rispetto a questo totale indipendentemente dal numero di file trasferiti per soddisfare la richiesta. Ciò significa che una richiesta di trasferimento che trasferisce un singolo file conta nello stesso modo di una richiesta di trasferimento che trasferisce 10 file.</p> <p>L'agent accoda i trasferimenti quando l'agente di destinazione raggiunge il limite specificato dalla proprietà Trasferimenti maxDestination.</p> <p>Se la somma dei seguenti valori di proprietà dell'agent: maxSourceTransfers + maxDestinationTransfers + maxQueuedTransfers supera il valore dell'impostazione MAXDEPTH della coda dell'archivio di stato (SYSTEM.FTE.STATE.<i>nome agent</i>), l'agent non viene avviato.</p>	<p>25 (per tutti gli agent tranne Connect:Direct)</p> <p>5 (per agent bridge Connect:Direct)</p>
maxFilesForTransfer	<p>Il numero massimo di elementi di trasferimento consentiti per un singolo trasferimento gestito. Se un trasferimento gestito contiene un numero di elementi superiore al valore di maxFilesForTransfer, il trasferimento gestito non riesce e non viene elaborato alcun elemento di trasferimento.</p> <p>L'impostazione di questa proprietà impedisce il trasferimento accidentale di troppi file a causa di una richiesta di trasferimento non corretta, ad esempio, se un utente specifica accidentalmente il trasferimento della directory root / sulla parola chiave conref=" ../common/mqent.dita#mqent/unixlinuxbis " /> sistemi.</p>	5000
Trasferimenti maxSource	<p>Il numero massimo di trasferimenti simultanei che l'agent di origine elabora in qualsiasi momento. Ogni richiesta di trasferimento inoltrata a un agente viene contata rispetto a questo totale indipendentemente dal numero di file trasferiti per soddisfare la richiesta. Ciò significa che una richiesta di trasferimento che trasferisce un singolo file conta nello stesso modo di una richiesta di trasferimento che trasferisce 10 file.</p> <p>Le code dell'agent di origine vengono trasferite quando l'agent di destinazione raggiunge il limite specificato dalla proprietà Trasferimenti maxSource.</p> <p>Se la somma dei seguenti valori di proprietà dell'agent: maxSourceTransfers + maxDestinationTransfers + maxQueuedTransfers supera il valore dell'impostazione MAXDEPTH della coda dell'archivio di stato (SYSTEM.FTE.STATE.<i>nome agent</i>), l'agent non viene avviato.</p>	<p>25 (per tutti gli agent tranne gli agent bridge Connect:Direct)</p> <p>5 (per agent bridge Connect:Direct)</p>
Trasferimenti maxQueued	<p>Il numero massimo di trasferimenti in sospeso che possono essere accodati da un agent di origine finché l'agent non rifiuta una nuova richiesta di trasferimento. È possibile impostare questa proprietà in modo che, nonostante i limiti di maxDestinationTransfers e maxSourceTransfers vengano soddisfatti o superati, tutte le nuove richieste di trasferimento effettuate ora vengono accettate, accodate e successivamente eseguite.</p> <p>L'ordine in cui vengono elaborate le richieste di trasferimento in coda è un fattore della loro priorità e per quanto tempo sono state accodate. I trasferimenti in sospeso vecchi e ad alta priorità vengono selezionati per primi. I trasferimenti con una priorità bassa che sono stati in coda per un lungo periodo di tempo vengono selezionati in preferenza ai trasferimenti con priorità più elevata e più recenti.</p> <p>Se la somma dei seguenti valori di proprietà dell'agent: maxSourceTransfers + maxDestinationTransfers + maxQueuedTransfers supera il valore dell'impostazione MAXDEPTH della coda dell'archivio di stato (SYSTEM.FTE.STATE.<i>nome agent</i>), l'agent non viene avviato.</p>	1000

Tabella 90. Proprietà agent avanzate: Timeout di ripristino trasferimento

Nome proprietà	Descrizione	Valore predefinito
Timeout transferRecovery	<p>Impostare la quantità di tempo, in secondi, durante la quale l'agent di origine continua a tentare di ripristinare un trasferimento file bloccato.</p> <p>Quando la proprietà non è impostata, il comportamento predefinito dell'agent è di continuare a riprovare fino a quando non recupera correttamente il trasferimento. È possibile impostare i seguenti valori per la proprietà di timeout di ripristino trasferimento:</p> <p>-1 L'agent continua a tentare di recuperare il trasferimento in stallo fino al completamento del trasferimento. L'uso di questa opzione equivale al comportamento predefinito dell'agente quando la proprietà non è impostata.</p> <p>0 L'agent arresta il trasferimento file non appena avvia il ripristino.</p> <p>>0 L'agent continua a tentare di recuperare il trasferimento in stallo per il periodo di tempo in secondi come impostato dal valore intero positivo specificato. Ad esempio, <code>transferRecoveryTimeout=21600</code> indica che l'agent continua a tentare di ripristinare il trasferimento per 6 ore da quando entra nel ripristino. Il valore massimo per questo parametro è 99999999.</p>	-1

Tabella 91. Proprietà avanzate dell'agent: routine di uscita utente



Nome proprietà	Descrizione	Valore predefinito
Delimitatori agentForceConsistentPath	Forzare il delimitatore di percorso nelle informazioni del file di origine e del file di destinazione fornite alle uscite di trasferimento in modo che sia lo stile UNIX : barra (/). Le opzioni valide sono <code>true</code> e <code>false</code> .	No
Classi destinationTransferEndExit	Specifica un elenco separato da virgole di classi che implementano una routine di uscita utente di trasferimento di destinazione.	Nessun valore predefinito
Classi destinationTransferStartExit	Specifica un elenco separato da virgole di classi che implementano una routine di uscita utente di inizio trasferimento di destinazione.	Nessun valore predefinito
Percorso exitClass	Specifica un elenco delimitato da caratteri, specifico della piattaforma, di indirizzari che fungono da percorso classi per le routine di uscita utente. La ricerca nella directory delle uscite dell'agent viene eseguita prima di qualsiasi voce in questo percorso di classe.	Directory exits dell'agente
exitNativeLibraryPath	Specifica un elenco delimitato da caratteri, specifico della piattaforma, di indirizzari che fungono da percorso della libreria nativa per le routine di uscita utente.	Directory exits dell'agente
ioMaxRecordLength	La lunghezza record massima, in byte, che può essere supportata per un file orientato ai record. Managed File Transfer può supportare la scrittura in file orientati ai record con qualsiasi lunghezza di record. Tuttavia, lunghezze record elevate potrebbero causare errori di memoria esaurita, quindi per evitare questi errori la lunghezza massima del record è limitata per impostazione predefinita a 64 K. Durante la lettura da file orientati ai record, un intero record deve essere contenuto in una singola porzione di trasferimento, quindi la lunghezza del record è ulteriormente limitata dalla dimensione della porzione di trasferimento. Questa proprietà viene utilizzata solo per i file orientati ai record dell'uscita utente I/O.	64 KB
monitorExitClasses	Specifica un elenco separato da virgole di classi che implementano una routine di uscita di controllo. Per ulteriori informazioni, consultare Uscite utente di monitoraggio delle risorse MFT .	Nessun valore predefinito
protocolBridgeCredentialExitClasses	Specifica un elenco separato da virgole di classi che implementano una routine uscita utente di credenziali del bridge di protocollo. Per ulteriori informazioni, consultare Associazione credenziali per un server file utilizzando le classi di uscita .	Nessun valore predefinito.
Classi sourceTransferEndExit	Specifica un elenco separato da virgole di classi che implementano una routine di uscita di fine trasferimento di origine.	Nessun valore predefinito
Classi sourceTransferStartExit	Specifica un elenco separato da virgole di classi che implementano una routine di uscita di inizio trasferimento di origine.	Nessun valore predefinito
IOExitClasses	Specifica un elenco separato da virgole di classi che implementano una routine uscita utente I/O. Elencare solo le classi che implementano l'interfaccia IOExit, ossia non elencare le classi che implementano le altre interfacce di uscita utente I/O, ad esempio IOExitResourcePath e IOExitChannel. Per ulteriori informazioni, consultare Utilizzo delle uscite utente I/O di trasferimento MFT .	Nessun valore predefinito.

Tabella 92. Proprietà agent avanzate: compressione client IBM MQ		
Nome proprietà	Descrizione	Valore predefinito
Compressione agentData	<p>Questa proprietà è supportata solo per connessioni client.</p> <p>Un elenco separato da virgole dei tipi di compressione per il trasferimento dei dati del file da negoziare con il server IBM MQ remoto. È possibile trovare informazioni su questi tipi di compressione nel seguente argomento: Elenco di compressione dei dati del messaggio.</p> <p>I valori vengono controllati per la validità e quindi trasmessi in ordine di aspetto come proprietà al canale del client agent. Il client IBM MQ gestisce quindi la negoziazione tra questo canale client e il canale server remoto per trovare il denominatore comune più basso corrispondente tra le proprietà di compressione sui due canali. Se non viene trovata alcuna corrispondenza, viene sempre selezionato MQCOMPRESS_NONE.</p>	MQCOMPRESS_NONE
Compressione agentHeader	<p>Questa proprietà è supportata solo per connessioni client.</p> <p>Un elenco separato da virgole dei tipi di compressione per il trasferimento dei dati di intestazione da negoziare con il server IBM MQ remoto. I valori accettati sono MQCOMPRESS_NONE o MQCOMPRESS_SYSTEM. È possibile trovare informazioni su questi tipi di compressione nel seguente argomento: HdrCompList [2] (MQLONG).</p> <p>I valori vengono controllati per la validità e quindi trasmessi in ordine di aspetto come proprietà al canale del client agent. Il client IBM MQ gestisce quindi la negoziazione tra questo canale client e il canale server remoto per trovare il denominatore comune più basso corrispondente tra le proprietà di compressione sui due canali. Se non viene trovata alcuna corrispondenza, viene sempre selezionato MQCOMPRESS_NONE.</p>	MQCOMPRESS_NONE

z/OS

Tabella 93. Proprietà avanzate dell'agent: specifiche di z/OS		
Nome proprietà	Descrizione	Valore predefinito
adminGroup	<p>Un gruppo gestore della sicurezza. I membri di questo gruppo possono:</p> <ul style="list-style-type: none"> • Avviare l'agent utilizzando il comando fteStartAgent . • Arrestare l'agent utilizzando il comando fteStopAgent . • Attivare o disattivare la traccia per l'agent utilizzando il comando fteSetAgentTraceLevel . • Attivare o disattivare i log per l'agent utilizzando il comando fteSetAgentLogLevel • Visualizzare i dettagli di un agent locale eseguendo il comando fteShowAgentDetails con il parametro -d specificato. <p>Definire un gruppo del gestore della sicurezza, ad esempio MFTADMIN , quindi aggiungere l'ID utente dell'attività avviata e gli ID TSO dell'amministratore a questo gruppo. Modificare il file delle proprietà dell'agente e impostare la proprietà adminGroup in modo che sia il nome di questo gruppo del gestore sicurezza.</p> <pre>adminGroup=MFTADMIN</pre>	Nessuna
z/OS bpxwdynAllocAdditionalOptions	<p>Managed File Transfer utilizza l'interfaccia di testo BPXWDYN per creare e aprire i dataset z/OS . Quando BPXWDYN viene utilizzato per l'assegnazione del dataset per impostazione predefinita, Managed File Transfer garantisce, quando possibile, che il dispositivo dati sia montato (non richiesto per i dataset basati su disco, ma richiesto per i dataset nastro). Poiché le opzioni potrebbero non essere supportate per determinati ambienti, utilizzare questa proprietà per modificare questo comportamento. Inoltre, quando si esegue il trasferimento a un dataset, è anche possibile specificare le opzioni per BPXWDYN sulla riga comandi; queste opzioni sono in aggiunta a quelle specificate da questa proprietà.</p> <p>Alcune opzioni BPXWDYN non devono essere specificate quando si utilizzano le proprietà bpxwdynAllocAdditionalOptions nel file <code>agent.properties</code> . Per un elenco di queste proprietà, consultare Proprietà BPXWDYN che non devono essere utilizzate con MFT.</p>	<p>Il valore predefinito è il seguente:</p> <ul style="list-style-type: none"> • MOUNT per z/OS V1R8 e successive
armELEMType	<p>Proprietà facoltativa. Se l'agent è configurato per il riavvio da parte di ARM (Automatic Restart Manager), impostare questa proprietà sul valore del parametro ARM ELEMType specificato nella politica ARM associata. Per un agent, impostare ELEMType su SYSBFAG.</p>	Non impostato

Tabella 93. Proprietà avanzate dell'agent: specifiche di z/OS (Continua)		
Nome proprietà	Descrizione	Valore predefinito
armELEMENT	Proprietà facoltativa. Se l'agent è configurato per il riavvio da ARM (Automatic Restart Manager), impostare questa proprietà sul valore del parametro ARM ELEMENT specificato nella politica ARM associata. È possibile impostare il valore ELEMENT in modo che corrisponda al nome dell'agent.	Non impostato

Tabella 94. Proprietà agent avanzate: Altre proprietà		
Nome proprietà	Descrizione	Valore predefinito
  legacyXMLMessageMQMDFormat	<p>Managed File Transfer I messaggi XML generati dall'agent (ad esempio, i messaggi di avanzamento del trasferimento e del log), vengono ora inviati ad una coda con un campo in formato MQMD vuoto. Le versioni precedenti del prodotto impostano il campo del formato MQMD su MQSTR (una stringa di messaggio di testo). L'impostazione di questa proprietà su true abilita l'invio dei messaggi XML Managed File Transfer generati dall'agent a una coda con il campo formato MQMD MQSTR.</p> <p>Nota: I messaggi di risposta dell'agente ai comandi verranno inviati con un formato di messaggio corrispondente alla richiesta di comando corrispondente.</p> <p>Se il campo del formato MQMD è impostato su MQSTR, è possibile che i messaggi XML del comando Managed File Transfer siano danneggiati se nella rete MQ sono presenti canali con la conversione dei dati abilitata.</p>	No
adjustScheduleTimeForDaylightSav ing	<p>Se l'azienda esegue i trasferimenti pianificati ogni giorno, perché il trasferimento pianificato è stato creato con:</p> <ul style="list-style-type: none"> • Parametro -oi impostato su giorni e • Parametro -tb impostato sull'origine <p>nel comando <code>fteCreateTransfer</code>, ad esempio, l'impostazione di questa proprietà su <code>true</code> sposterà il tempo di trasferimento pianificato in avanti di un'ora quando gli orologi vanno avanti di un'ora e indietro di un'ora quando gli orologi tornano indietro di un'ora.</p> <p>Ad esempio, se il trasferimento pianificato deve essere eseguito all' 1:00, quando gli orologi vanno avanti, il trasferimento verrà eseguito alle 2:00 e quando gli orologi tornano indietro, il trasferimento ritorna all' 1:00.</p>	vero, true

Concetti correlati

[Opzioni di configurazione MFT su Multiplatforms](#)

[Opzione di timeout per i trasferimenti file in recupero](#)

[MFT sandboxes](#)

Attività correlate

[Configurazione di un agent MFT per più canali in un cluster](#)

[Configurazione di un agent MFT per più canali: non in cluster](#)

Riferimenti correlati

[“Java proprietà di sistema per MFT” a pagina 220](#)

Un numero di proprietà del comando e dell'agent Managed File Transfer deve essere definito come proprietà di sistema Java, poiché definiscono la configurazione per la funzione precedente che non è in grado di utilizzare il meccanismo delle proprietà del comando o dell'agent.

[Proprietà SSL/TLS per MFT](#)

[“File MFT command.properties” a pagina 203](#)

Il file `command.properties` specifica il gestore code comandi a cui connettersi quando si immettono i comandi e le informazioni richieste da Managed File Transfer per contattare tale gestore code.

[“File MFT coordination.properties” a pagina 199](#)

Il file `coordination.properties` specifica i dettagli di collegamento al gestore code di coordinamento. Poiché diverse installazioni di Managed File Transfer potrebbero condividere lo stesso gestore code di coordinamento, è possibile utilizzare un link simbolico a un file `coordination.properties` comune su un'unità condivisa.

[“File MFT logger.properties” a pagina 207](#)

Il programma di registrazione Managed File Transfer ha una serie di proprietà di configurazione. Specificare queste propriet ... nel file `logger.properties`, che si trova nella directory `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

[Agent fteCreate](#)

[fteCreateBridgeAgent](#)

[CDAgent fteCreate](#)

[“L'utilizzo delle variabili di ambiente nelle proprietà MFT” a pagina 169](#)

È possibile utilizzare le variabili di ambiente nelle proprietà Managed File Transfer che rappresentano ubicazioni di file o directory. Ciò consente alle ubicazioni dei file o delle directory utilizzate durante l'esecuzione di parti del prodotto di variare in base all'ambiente corrente (ad esempio, l'utente che esegue un comando).

File MFT coordination.properties

Il file `coordination.properties` specifica i dettagli di collegamento al gestore code di coordinamento. Poiché diverse installazioni di Managed File Transfer potrebbero condividere lo stesso gestore code di coordinamento, è possibile utilizzare un link simbolico a un file `coordination.properties` comune su un'unità condivisa.

Il file `coordination.properties` viene creato dal programma di installazione o dal comando **fteSetupCoordination**. È possibile utilizzare il comando **fteSetupCoordination** con l'indicatore **-f** per cambiare le proprietà del gestore code di coordinamento di base in questo file. Per modificare o aggiungere le proprietà avanzate del gestore code di coordinamento, è necessario modificare il file in un editor di testo.

Il file `coordination.properties` si trova nella directory `MQ_DATA_PATH/mqft/config/coordination_qmgr_name`.

Il file MFT `coordination.properties` contiene i seguenti valori:

Tabella 95. Proprietà del gestore code di coordinamento di base		
Nome proprietà	Descrizione	Valore predefinito
<code>coordinationCredentialsKeyFile</code>	Nome del file contenente la chiave della credenziale utilizzata durante la codifica delle credenziali.	Una proprietà stringa che non ha un valore predefinito.
<code>coordinationQMgr</code>	Il nome del gestore code di coordinamento.	Nessun valore predefinito
<code>Host coordinationQMgr</code>	Il nome host o l'indirizzo IP del gestore code di coordinamento.	Nessun valore predefinito
<code>Porta coordinationQMgr</code>	Il numero di porta utilizzato per le connessioni client al gestore code di coordinamento.	1414
<code>Canale coordinationQMgr</code>	Il nome del canale SVRCONN utilizzato per connettersi al gestore code di coordinamento.	SYSTEM.DEF.SVRCONN

Se non si specifica un valore per la proprietà `Host coordinationQMgr`, per impostazione predefinita viene utilizzata la modalità di bind.

Se si specifica un valore per la proprietà `Host coordinationQMgr` ma non si specificano i valori per le proprietà `coordinationQMgrPort` e `coordinationQMgrChannel`, un numero di porta 1414 e un canale `SYSTEM.DEF.SVRCONN` vengono utilizzati per impostazione predefinita.

Di seguito è riportato un esempio del contenuto di un file `coordination.properties`:

```
coordinationQMgr=ERIS
coordinationQMgrHost=kuiper.example.com
coordinationQMgrPort=2005
coordinationQMgrChannel=SYSTEM.DEF.SVRCONN
```

In questo esempio, ERIS è il nome di un gestore code IBM MQ ubicato sul sistema `kuiper.example.com`. Il gestore code ERIS è il gestore code a cui Managed File Transfer invia le informazioni di log.

Proprietà di coordinamento avanzato

Managed File Transfer fornisce anche proprietà di coordinamento più avanzate. Se si desidera utilizzare una delle seguenti proprietà, modificare manualmente il file `coordination.properties` per aggiungere le proprietà avanzate richieste. Le parentesi, le virgole (,) e le barre retroverse (\) sono caratteri speciali nei comandi MFT e devono essere preceduti da un carattere barra retroversa (\).

Windows I percorsi dei file su Windows possono essere specificati utilizzando doppie barre rovesciate (\\) come separatore o utilizzando singole barre (/). Per ulteriori informazioni sull'escape dei caratteri nei file delle proprietà Java, consultare la documentazione di Oracle [Javadoc](#) per la classe `Proprietà`.

- [Proprietà agente](#)
- [Proprietà codepage](#)
- [proprietà di connessione](#)
- [Proprietà del gestore code a più istanze](#)
- [Proprietà delle code](#)
- [Proprietà di sicurezza](#)
- [Proprietà SSL](#)
- [Proprietà delle sottoscrizioni](#)

Tabella 96. Proprietà di coordinamento avanzate: Agent

Nome proprietà	Descrizione	Valore predefinito
agentStatusJitterTolerance	La quantità massima di tempo per cui una pubblicazione del messaggio di stato dell'agent può essere ritardata prima che il messaggio venga considerato scaduto. Questo valore viene misurato in millisecondi. L'età di un messaggio di stato si basa sull'ora in cui è stato pubblicato nel gestore code di coordinamento. Tuttavia, il messaggio viene emesso dall'agent qualche tempo prima che venga ricevuto nel gestore code di coordinamento per consentire il tempo richiesto per viaggiare attraverso la rete IBM MQ. Se questo transito impiega sempre la stessa quantità di tempo, i messaggi creati a distanza di 60 secondi vengono pubblicati a distanza di 60 secondi, indipendentemente dal tempo effettivo in transito. Tuttavia, se il tempo di transito varia tra i messaggi, potrebbero essere creati a intervalli di 60 secondi ma pubblicati a intervalli di, ad esempio, 61, 59, 58 e 62 secondi. La deviazione massima da 60, 2 secondi in questo esempio, è il jitter. Questa proprietà determina il ritardo massimo dovuto al jitter prima che il messaggio venga considerato scaduto.	3000

Tabella 97. Proprietà di coordinamento avanzate: Code page

Nome proprietà	Descrizione	Valore predefinito
coordinationCcsid	La codepage con cui i comandi si collegano al gestore code di coordinamento. Inoltre, tutte le pubblicazioni sul gestore code di coordinamento effettuate dall'agent vengono eseguite con questa codepage. Se si specifica un valore per <code>coordinationCcsid</code> è necessario specificare anche un valore per il nome <code>coordinationCcsid</code> .	1208
Nome <code>coordinationCcsid</code>	La rappresentazione Java di <code>coordinationCcsid</code> . Se si specifica un valore per il nome <code>coordinationCcsid</code> , è necessario specificare anche un valore per <code>coordinationCcsid</code> .	UTF8

Tabella 98. Proprietà di coordinamento avanzate: Connessione

Nome proprietà	Descrizione	Valore predefinito
Percorso <code>javaLibrary</code>	Quando ci si connette a un gestore code in modalità di bind, Managed File Transfer deve avere accesso alle librerie di bind IBM MQ Java. Per impostazione predefinita, Managed File Transfer cerca le librerie di bind nell'ubicazione predefinita definita da IBM MQ. Se le librerie di bind si trovano in un'ubicazione differente, utilizzare questa proprietà per specificare l'ubicazione delle librerie di bind.	<code>MQ_INSTALLATION_PATH/</code> <code>java/lib</code>

Tabella 99. Proprietà di coordinamento avanzate: gestore code a più istanze		
Nome proprietà	Descrizione	Valore predefinito
coordinationQMGrStandby	Il nome host e il numero di porta utilizzati per connessioni client, in formato IBM MQ CONNAME, per l'istanza in standby di un gestore code di coordinamento a più istanze definito dalla proprietà coordinationQMGr . Ad esempio, <i>host_name(port_number)</i>	Nessun valore predefinito

Tabella 100. Proprietà di coordinamento avanzate: Coda		
Nome proprietà	Descrizione	Valore predefinito
Prefisso dynamicQueue	Questa proprietà definisce il prefisso IBM MQ da utilizzare per generare un nome coda temporaneo. Il formato della proprietà Prefisso dynamicQueue segue il formato del campo DynamicQName della struttura MQOD IBM MQ . Per ulteriori informazioni, consultare Creazione di code dinamiche . È anche possibile definire questa proprietà nel file <code>command.properties</code> se si desidera utilizzare un prefisso IBM MQ specifico per le code di risposta temporanee generate dai comandi che richiedono una risposta dall'agente.	WMQFTE.*
Nome modelQueue	Questa proprietà definisce la coda modello IBM MQ da utilizzare per la generazione di una coda temporanea. È possibile definire questa proprietà anche nel file <code>command.properties</code> se si desidera utilizzare una specifica coda modello IBM MQ per le code di risposta temporanee generate dai comandi che richiedono una risposta dall'agent. Per ulteriori informazioni, consultare " File MFT command.properties " a pagina 203.	SYSTEM.DEFAULT.MODEL.QUEUE




Tabella 101. Proprietà di coordinamento avanzate: Sicurezza		
Nome proprietà	Descrizione	Valore predefinito
userIdForClientConnect	L'ID utente che viene passato attraverso le connessioni client a IBM MQ. Se si specifica <i>java</i> , il nome utente riportato dalla JVM viene fornito come parte della richiesta di connessione IBM MQ . Il valore di questa proprietà può essere Nessuno o <i>java</i> .	Nessuna
File coordinationQMGrAuthenticationCredentials	Il percorso al file che contiene le credenziali di connessione di MQ per la connessione al gestore code di coordinamento.	 Per i dettagli sulla creazione del File delle credenziali di autenticazione, consultare Configurazione di MQMFTCredentials.xml su z/OS .  Fare riferimento a Configurazione di MQMFTCredentials.xml per informazioni sull'ubicazione e le autorizzazioni di questo file.  Ulteriori dettagli sulla creazione del file delle credenziali di autenticazione si trovano in MFT e nell' autenticazione della connessione IBM MQ

Tabella 102. Proprietà di coordinamento avanzate: SSL/TLS		
Nome proprietà	Descrizione	Valore predefinito
coordinationSslCipherSpec	Specifica il protocollo, l'algoritmo hash e l'algoritmo di codifica utilizzato e il numero di bit utilizzati nella chiave di codifica, quando i dati vengono scambiati tra i comandi e il gestore code di coordinamento. Il valore di <code>coordinationSslCipherSpec</code> è un nome CipherSpec . Questo nome CipherSpec è uguale al nome CipherSpec utilizzato nel canale del gestore code di coordinamento. Un elenco di nomi CipherSpec è incluso in <code>CipherSpec CipherSpecs</code> e <code>CipherSuites</code> nelle classi IBM MQ per Java e <code>SSL/TLS CipherSpecs</code> e <code>CipherSuites</code> nelle classi IBM MQ per JMS. <code>coordinationSslCipherSpec</code> è simile a <code>coordinationSslCipherSuite</code> . Se vengono specificati sia <code>coordinationSslCipherSuite</code> che <code>coordinationSslCipherSpec</code> , viene utilizzato il valore di <code>coordinationSslCipherSpec</code> .	Nessuna

Tabella 102. Proprietà di coordinamento avanzate: SSL/TLS (Continua)

Nome proprietà	Descrizione	Valore predefinito
coordinationSslCipherSuite	<p>Specifica gli aspetti SSL della modalità di scambio dei dati tra i comandi e il gestore code di coordinamento.</p> <p>Il valore di coordinationSslCipherSuite è un nome CipherSuite . Il nome CipherSuite corrisponde al nome CipherSpec utilizzato sul canale gestore code dell'agent. Per ulteriori informazioni, consultare CipherSuite e le associazioni di nomi CipherSpec.</p> <p>coordinationSslCipherSuite è simile a coordinationSslCipherSpec. Se vengono specificati sia coordinationSslCipherSuite che coordinationSslCipherSpec , viene utilizzato il valore di coordinationSslCipherSpec .</p>	Nessuna
coordinationSslPeerName	<p>Specifica una struttura di nomi distinti che deve corrispondere al nome fornito dal gestore code di coordinamento. Il DN (distinguished name) viene utilizzato per controllare il certificato di identificazione presentato dal gestore code di coordinamento sulla connessione.</p>	Nessuna
coordinationSslTrustStore	<p>Specifica l'ubicazione dei certificati ritenuti attendibili dai comandi. Il valore di coordinationSslTrustStore è un percorso file. Le parentesi, le virgole (,) e le barre retroverse (\) sono caratteri speciali nei comandi MFT e devono essere preceduti da un carattere barra retroversa (\).</p> <p>Windows I percorsi dei file su Windows possono essere specificati utilizzando doppie barre rovesciate (\\) come separatore o utilizzando singole barre (/).</p> <p>Da IBM WebSphere MQ 7.5 o versioni successive, il valore di questa proprietà può contenere variabili di ambiente.</p>	Nessuna
coordinationSslTrustStore	<p>Il tipo di keystore SSL che si desidera utilizzare. Sono supportati i keystore JKS e PKCS#12 . Il valore di questa proprietà può essere jks o pkcs12.</p>	JKS
coordinationSslTrustStoreCredentialsFile	<p>Il percorso al file che contiene le credenzialiTrustStore coordinationSsl.</p> <p>Il valore di questa proprietà può contenere variabili di ambiente.</p>	Il valore predefinito per questa proprietà è %USERPROFILE%/MQMFTCredentials.xml su Windows e \$HOME/MQMFTCredentials.xml su altre piattaforme.
coordinationSslKeyStore	<p>Specifica l'ubicazione della chiave privata dei comandi. Il valore di coordinationSslKeyStore è un percorso file. Le parentesi, le virgole (,) e le barre retroverse (\) sono caratteri speciali nei comandi MFT e devono essere preceduti da un carattere barra retroversa (\).</p> <p>Windows I percorsi dei file su Windows possono essere specificati utilizzando doppie barre rovesciate (\\) come separatore o utilizzando singole barre (/). Questa proprietà è richiesta solo se il gestore code di coordinamento richiede l'autenticazione client.</p> <p>Il valore di questa proprietà può contenere variabili di ambiente.</p>	Nessuna
Tipo coordinationSslKeyStore	<p>Il tipo di keystore SSL che si desidera utilizzare. Sono supportati i keystore JKS e PKCS#12 . Il valore di questa proprietà può essere jks o pkcs12.</p>	JKS
coordinationSslKeyStoreCredentialsFile	<p>Il percorso del file che contiene le credenziali coordinationSslKeyStore .</p> <p>Il valore di questa proprietà può contenere variabili di ambiente.</p>	Il valore predefinito per questa proprietà è %USERPROFILE%/MQMFTCredentials.xml su Windows e \$HOME/MQMFTCredentials.xml su altre piattaforme.
coordinationSslFipsRequired	<p>Specifica che si desidera abilitare il supporto FIPS al livello del gestore code di coordinamento. Il valore di questa proprietà può essere true o false. Per ulteriori informazioni, consultare Supporto FIPS in MFT.</p>	No

Tabella 103. Proprietà di coordinamento avanzato: Sottoscrizione

Nome proprietà	Descrizione	Valore predefinito
Argomento coordinationSubscription	<p>Utilizzare questa proprietà per specificare un argomento diverso da SYSTEM.FTE a cui sottoscrivere per ottenere le pubblicazioni sullo stato della rete IBM MQ . Tutti gli strumenti vengono ancora pubblicati nel SISTEMA SYSTEM.FTE , ma è possibile modificare la topologia IBM MQ per distribuire queste pubblicazioni a diversi argomenti in base ai relativi contenuti. Puoi quindi utilizzare questa funzione per forzare gli strumenti a sottoscrivere uno di questi altri argomenti.</p>	SYSTEM.FTE

Concetti correlati

[Opzioni di configurazione MFT su Multiplatforms](#)

Riferimenti correlati

[Coordinamento fteSetup](#)

[Proprietà SSL/TLS per MFT](#)

[“Il file MFT agent.properties” a pagina 175](#)

Ogni Managed File Transfer Agent ha il proprio file delle proprietà, `agent.properties`, che deve contenere le informazioni che un agente utilizza per connettersi al proprio gestore code. Il file `agent.properties` può anche contenere proprietà che modificano il comportamento dell'agent.

[“File MFT command.properties” a pagina 203](#)

Il file `command.properties` specifica il gestore code comandi a cui connettersi quando si immettono i comandi e le informazioni richieste da Managed File Transfer per contattare tale gestore code.

[“File MFT logger.properties” a pagina 207](#)

Il programma di registrazione Managed File Transfer ha una serie di proprietà di configurazione. Specificare queste propriet ... nel file `logger.properties`, che si trova nella directory `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

File MFT command.properties

Il file `command.properties` specifica il gestore code comandi a cui connettersi quando si immettono i comandi e le informazioni richieste da Managed File Transfer per contattare tale gestore code.

Il file `command.properties` viene creato dal programma di installazione o dal comando **fteSetupCommands**. È possibile utilizzare il comando **fteSetupCommands** con l'indicatore **-f** per modificare le proprietà di base del gestore code comandi in questo file. Per modificare o aggiungere le proprietà avanzate del gestore code comandi, è necessario modificare il file in un editor di testo.

Alcuni comandi Managed File Transfer si connettono al gestore code dell'agent o al gestore code di coordinamento invece che al gestore code dei comandi. Per informazioni su quali comandi si connettono a quale gestore code, consultare [Quale MFT comando si connette a quale gestore code](#).

Il file `command.properties` si trova nella directory `MQ_DATA_PATH/mqft/config/coordination_qmgr_name`.

Il file MFT `command.properties` contiene i seguenti valori:

Nome proprietà	Descrizione	Valore predefinito
<code>connectionCredentialsKeyFile</code>	Nome del file contenente la chiave della credenziale utilizzata durante la codifica delle credenziali.	Una proprietà stringa che non ha un valore predefinito.
<code>connectionQMGr</code>	Il nome del gestore code utilizzato per connettersi alla rete IBM MQ.	Nessun valore predefinito
<code>Host connectionQMGr</code>	Il nome host o l'indirizzo IP del gestore code di connessione.	Nessun valore predefinito
<code>Porta connectionQMGr</code>	Il numero di porta utilizzato per connettersi al gestore code di connessione in modalità client.	1414
<code>Canale connectionQMGr</code>	Il nome del canale SVRCONN utilizzato per connettersi al gestore code di connessione.	SYSTEM.DEF.SVRCONN

Se non si specifica un valore per la proprietà `Host connectionQMGr`, per impostazione predefinita viene utilizzata la modalità di bind.


Se si specifica un valore per la proprietà `connectionQMGrHost` ma non si specificano valori per le proprietà `connectionQMGrPort` e `connectionQMGrChannel`, un numero di porta 1414 e un canale `SYSTEM.DEF.SVRCONN` vengono utilizzati per impostazione predefinita.

Di seguito è riportato un esempio del contenuto di un file `command.properties`:

```
connectionQMGr=PLUTO
connectionQMGrHost=kuiper.example.com
connectionQMGrPort=1930
connectionQMGrChannel=SYSTEM.DEF.SVRCONN
```

In questo esempio, PLUTO è il nome di un gestore code IBM MQ ubicato sul sistema `kuiper.example.com`. Il gestore code PLUTO è il gestore code a cui si connettono i comandi Managed File Transfer .

Proprietà avanzate del comando

Managed File Transfer fornisce anche proprietà di comandi più avanzate. Se si desidera utilizzare una delle seguenti proprietà, modificare manualmente il file `command.properties` per aggiungere le proprietà avanzate richieste. Le parentesi, le virgole (,) e le barre retroverse (\) sono caratteri speciali nei comandi MFT e devono essere preceduti da un carattere barra roversa (\).  I percorsi dei file su Windows possono essere specificati utilizzando doppie barre rovesciate (\\) come separatore o utilizzando singole barre (/). Per ulteriori informazioni sull'escape dei caratteri nei file delle proprietà Java, consultare la documentazione di Oracle [Javadoc](#) per la classe `Proprietà`.

- [Proprietà agente](#)
- [Proprietà codepage](#)
- [Proprietà del gestore code a più istanze](#)
- [Proprietà delle code](#)
- [Proprietà di sicurezza](#)
- [Proprietà SSL](#)

Nome proprietà	Descrizione	Valore predefinito
<code>failCleanAgentWithNoArguments</code>	Per impostazione predefinita, il valore di questa proprietà è <code>true</code> , il che significa che l'esecuzione del comando fteCleanAgent non riesce se viene specificato solo il parametro del nome <code>agent</code> . L'impostazione della proprietà su <code>false</code> significa che, se è impostato solo il parametro del nome dell' <code>agent</code> , il comportamento del comando fteCleanAgent è equivalente alla specifica del parametro -all .	vero, <code>true</code>

Nome proprietà	Descrizione	Valore predefinito
<code>connectionCcsid</code>	La codepage con cui i comandi si connettono al gestore code comandi. Se si specifica un valore per <code>connectionCcsid</code> è necessario specificare anche un valore per il nome <code>connectionCcsid</code> .	1208
Nome <code>connectionCcsid</code>	La rappresentazione Java di <code>connectionCcsid</code> . Se si specifica un valore per il nome <code>connectionCcsid</code> , è necessario specificare anche un valore per <code>connectionCcsid</code> .	UTF8

Nome proprietà	Descrizione	Valore predefinito
<code>connectionQMGrStandby</code>	Il nome host e il numero di porta utilizzati per le connessioni client, in formato IBM MQ CONNAME, per l'istanza in standby di un gestore code a più istanze definito dalla proprietà <code>connectionQMGr</code> . Ad esempio, <code>host_name(port_number)</code>	Nessun valore predefinito

Nome proprietà	Descrizione	Valore predefinito
Prefisso <code>dynamicQueue</code>	Per i comandi che richiedono una risposta dall' <code>agent</code> , questa proprietà definisce il prefisso IBM MQ da utilizzare per generare il nome della coda di risposta temporanea. Il formato della proprietà <code>Prefisso dynamicQueue</code> segue il formato del campo DynamicQName della struttura MQOD IBM MQ. Per ulteriori informazioni, consultare Creazione di code dinamiche . È inoltre possibile definire questa proprietà nel file <code>coordination.properties</code> se si desidera utilizzare un prefisso IBM MQ specifico per code temporanee generate da WMQFTE.	WMQFTE.*





Tabella 108. Proprietà del comando avanzate: Coda (Continua)		
Nome proprietà	Descrizione	Valore predefinito
Nome modelQueue	Per i comandi che richiedono una risposta dall'agent, questa proprietà definisce la coda modello IBM MQ da utilizzare per generare la coda di risposta temporanea. È inoltre possibile definire questa proprietà nel file coordination.properties se si desidera utilizzare una specifica coda modello IBM MQ per le code temporanee generate da WMQFTE. Per ulteriori informazioni, consultare "File MFT coordination.properties" a pagina 199.	SYSTEM.DEFAULT.MODEL.QUEUE
Proprietà connessione:		
Percorso javaLibrary	Quando ci si connette a un gestore code in modalità di bind, Managed File Transfer deve avere accesso alle librerie di bind IBM MQ Java . Per impostazione predefinita, Managed File Transfer cerca le librerie di bind nell'ubicazione predefinita definita da IBM MQ. Se le librerie di bind si trovano in un'ubicazione differente, utilizzare questa proprietà per specificare l'ubicazione delle librerie di bind.	/opt/mqm/java/lib
  legacyXMLMessageMQMDFormat	I messaggi XML di comandi Managed File Transfer vengono ora inviati a una coda con un campo di formato MQMD vuoto. Le versioni precedenti del prodotto impostano il campo del formato MQMD su MQSTR (una stringa di messaggio di testo). L'impostazione di questa proprietà su true consente ai messaggi XML del comando Managed File Transfer di essere inviati a una coda con il campo del formato MQMD di MQSTR. Se il campo del formato MQMD è impostato su MQSTR, è possibile che i messaggi XML del comando Managed File Transfer siano danneggiati se nella rete MQ sono presenti canali con la conversione dei dati abilitata.	No

Tabella 109. Proprietà del comando avanzate: Sicurezza		
Nome proprietà	Descrizione	Valore predefinito
userIdForClientConnect	L'ID utente che viene passato attraverso le connessioni client a IBM MQ. Se si specifica <i>java</i> , il nome utente riportato dalla JVM viene fornito come parte della richiesta di connessione IBM MQ . Il valore di questa proprietà può essere Nessuno o java.	Nessuna
connectionQMGrAuthenticationCredentialsAuthenticationCredentials	Il percorso del file che contiene le credenziali di connessione di MQ per la connessione al gestore code comandi.	Consultare Autenticazione della connessioneMFT e IBM MQ e i relativi argomenti secondari.

Tabella 110. Proprietà avanzate del comando: SSL/TLS		
Nome proprietà	Descrizione	Valore predefinito
connectionSslCipherSpec	Specifica il protocollo, l'algoritmo hash e l'algoritmo di crittografia utilizzato e quanti bit vengono utilizzati nella chiave di crittografia, quando i dati vengono scambiati tra i comandi e il gestore code comandi. Il valore di connectionSslCipherSpec è un nome CipherSpec . Questo nome CipherSpec è uguale al nome CipherSpec utilizzato sul canale del gestore code comandi. Un elenco di nomi CipherSpec validi è incluso in SSL/TLS CipherSpecs e CipherSuites nelle classi IBM MQ per Java e SSL/TLS CipherSpecs e CipherSuites nelle classi IBM MQ per JMS . connectionSslCipherSpec è simile a connectionSslCipherSuite. Se vengono specificati sia connectionSslCipherSuite che connectionSslCipherSpec , viene utilizzato il valore di connectionSslCipherSpec .	Nessuna
connectionSslCipherSuite	Specifica gli aspetti SSL del modo in cui i comandi e il gestore code comandi si scambiano dati. Il valore di connectionSslCipherSuite è un nome CipherSuite . Il nome CipherSuite corrisponde al nome CipherSpec utilizzato sul canale gestore code dell'agent. Per ulteriori informazioni, consultare CipherSuite e le associazioni di nomi CipherSpec . connectionSslCipherSuite è simile a connectionSslCipherSpec. Se vengono specificati sia connectionSslCipherSuite che connectionSslCipherSpec , viene utilizzato il valore di connectionSslCipherSpec .	Nessuna
connectionSslPeerName	Specifica una struttura di nomi distinti che deve corrispondere al nome fornito dal gestore code comandi. Il DN (Distinguished Name) viene utilizzato per verificare il certificato di identificazione presentato dal gestore code comandi sulla connessione.	Nessuna

Tabella 110. Proprietà avanzate del comando: SSL/TLS (Continua)

Nome proprietà	Descrizione	Valore predefinito
connectionSslTrustStore	Specifica l'ubicazione dei certificati ritenuti attendibili dai comandi. Il valore di connectionSslTrustStore è un percorso file. Le parentesi, le virgole (,) e le barre retroverse (\) sono caratteri speciali nei comandi MFT e devono essere preceduti da un carattere barra retroversa (\).  I percorsi dei file su Windows possono essere specificati utilizzando doppie barre rovesciate (\\) come separatore o utilizzando singole barre (/). Il valore di questa proprietà può contenere variabili di ambiente.	Nessuna
Tipo connectionSslTrustStore	Il tipo di truststore SSL che si desidera utilizzare. Sono supportati i keystore JKS e PKCS#12. Il valore di questa proprietà può essere jks o pkcs12.	JKS
connectionSslTrustStoreCredentialsFile	Il percorso del file contenente le credenziali connectionSslTrustStore. Il valore di questa proprietà può contenere variabili di ambiente.	Il valore predefinito per questa proprietà è %USERPROFILE%/MQMFTCredentials.xml su Windows e \$HOME/MQMFTCredentials.xml su altre piattaforme.
connectionSslKeyStore	Specifica l'ubicazione della chiave privata dei comandi. Il valore di connectionSslKeyStore è un percorso file. Le parentesi, le virgole (,) e le barre retroverse (\) sono caratteri speciali nei comandi MFT e devono essere preceduti da un carattere barra retroversa (\).  I percorsi dei file su Windows possono essere specificati utilizzando doppie barre rovesciate (\\) come separatore o utilizzando singole barre (/). Questa proprietà è richiesta solo se il gestore code comandi richiede l'autenticazione client. Il valore di questa proprietà può contenere variabili di ambiente.	Nessuna
Tipo connectionSslKeyStore	Il tipo di keystore SSL che si desidera utilizzare. Sono supportati i keystore JKS e PKCS#12. Il valore di questa proprietà può essere jks o pkcs12. Il valore di questa proprietà può contenere variabili di ambiente.	JKS
connectionSslKeyStoreCredentialsFile	Il percorso del file che contiene le credenzialiKeyStore connectionSsl. Il valore di questa proprietà può contenere variabili di ambiente.	Il valore predefinito per questa proprietà è %USERPROFILE%/MQMFTCredentials.xml su Windows e \$HOME/MQMFTCredentials.xml su altre piattaforme.
connectionSslFipsRequired	Specifica che si desidera abilitare il supporto FIPS al livello del gestore code comandi. Il valore di questa proprietà può essere true o false. Per ulteriori informazioni, consultare Supporto FIPS in MFT .	No

Concetti correlati

[Opzioni di configurazione MFT su Multiplatforms](#)

Riferimenti correlati

[“Java proprietà di sistema per MFT” a pagina 220](#)

Un numero di proprietà del comando e dell'agent Managed File Transfer deve essere definito come proprietà di sistema Java, poiché definiscono la configurazione per la funzione precedente che non è in grado di utilizzare il meccanismo delle proprietà del comando o dell'agent.

Proprietà SSL/TLS per MFT

[“Il file MFT agent.properties” a pagina 175](#)

Ogni Managed File Transfer Agent ha il proprio file delle proprietà, agent.properties, che deve contenere le informazioni che un agente utilizza per connettersi al proprio gestore code. Il file agent.properties può anche contenere proprietà che modificano il comportamento dell'agent.

[“File MFT coordination.properties” a pagina 199](#)

Il file coordination.properties specifica i dettagli di collegamento al gestore code di coordinamento. Poiché diverse installazioni di Managed File Transfer potrebbero condividere lo stesso gestore code di coordinamento, è possibile utilizzare un link simbolico a un file coordination.properties comune su un'unità condivisa.

[“File MFT logger.properties” a pagina 207](#)

Il programma di registrazione Managed File Transfer ha una serie di proprietà di configurazione. Specificare queste propriet ... nel file `logger.properties`, che si trova nella directory `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.


Comandi `fteSetup`: creare il file MFT `command.properties`

`fteCleanAgent`: ripulitura di un agent MFT

File MFT `logger.properties`

Il programma di registrazione Managed File Transfer ha una serie di proprietà di configurazione. Specificare queste propriet ... nel file `logger.properties`, che si trova nella directory `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

È possibile utilizzare le variabili di ambiente in alcune proprietà Managed File Transfer che rappresentano le ubicazioni di file o directory. Ciò consente ai percorsi dei file o delle directory utilizzati durante l'esecuzione di parti del prodotto, di variare in base alle modifiche dell'ambiente, ad esempio l'utente che sta eseguendo il processo. Per ulteriori informazioni, consultare [“L'utilizzo delle variabili di ambiente nelle proprietà MFT”](#) a pagina 169.

Nota: Le parentesi, le virgole (,) e le barre retroverse (\) sono caratteri speciali nei comandi MFT e devono essere preceduti da un carattere barra retroversa (\).  I percorsi dei file su Windows possono essere specificati utilizzando doppie barre rovesciate (\\) come separatore o utilizzando singole barre (/). Per ulteriori informazioni sull'escape dei caratteri nei Java file delle propriet ... in Oracle, consultare Javadoc per la classe delle propriet ...

Il file MFT `logger.properties` contiene i seguenti valori:

- [“Proprietà di collegamento della modalità di collegamento”](#) a pagina 207
- [“Proprietà di connessione SSL/TLS della modalità client”](#) a pagina 215

Proprietà di collegamento della modalità di collegamento

Tabella 111. Proprietà di collegamento della modalità di collegamento per il file <code>logger.properties</code>		
Nome proprietà	Descrizione	Valore predefinito
<code>wmqfte.logger.type</code>	Il tipo di logger in uso: file o database. Impostare questo valore su FILE o DATABASE.	Nessun valore predefinito
<code>wmqfte.max.transaction.messages</code>	Il numero massimo di messaggi elaborati in una transazione prima del commit della transazione. In modalità di registrazione circolare, un gestore code ha una quantità fissa di spazio disponibile per i dati in corso. Accertarsi di impostare questa proprietà con un valore sufficientemente basso in modo che lo spazio disponibile non si esaurisca.	50
<code>wmqfte.max.transaction.time</code>	L'intervallo di tempo massimo in millisecondi che intercorre tra i commit delle transazioni.	5000
<code>wmqfte.max.consecutive.reject</code>	Il numero massimo di messaggi che possono essere rifiutati consecutivamente (vale a dire, senza incontrare un messaggio valido). Se questo numero viene superato, il programma di registrazione conclude che il problema non è con i messaggi stessi, ma con la configurazione. Ad esempio, se si rende una colonna nome - agent nel database più stretta di tutti i propri nomi agent, tutti i messaggi che fanno riferimento agli agent vengono rifiutati.	50

Tabella 111. Proprietà di collegamento della modalità di collegamento per il file `logger.properties` (Continua)




Nome proprietà	Descrizione	Valore predefinito
<code>wmqfte.reject.queue.name</code>	Il nome di una coda in cui il programma di registrazione inserisce i messaggi che il programma di registrazione non può gestire. Se si dispone di un programma di registrazione database, consultare MFT Gestione degli errori del programma di registrazione e rifiuto per i dettagli sui messaggi che potrebbero essere inseriti in questa coda.	<code>SYSTEM.FTE.LOG.RJCT.nome_programma di registrazione</code>
<code>wmqfte.command.queue.name</code>	Il nome di una coda da cui il programma di registrazione legge i messaggi di comando che controllano il funzionamento.	<code>SYSTEM.FTE.LOG.CMD.nome_programma di registrazione</code>
<code>wmqfte.queue.manager</code>	Il gestore code a cui si connette il programma di registrazione. Questo parametro è obbligatorio ed è tutto ciò che è necessario per le connessioni in modalità di bind al gestore code. (Per le proprietà per la connessione a un gestore code remoto, consultare Tabella 112 a pagina 215.)	Nessun valore predefinito
<code>wmqfte.message.source.type</code>	<p>Uno dei seguenti valori:</p> <p>registrazione automatica Il valore predefinito. Il programma di registrazione crea e utilizza la propria sottoscrizione gestita e durevole sul gestore code definito in <code>SYSTEM.FTE/Log/#</code>. Questo è un valore appropriato per la maggior parte degli scenari.</p> <p>sottoscrizione amministrativa Se la sottoscrizione automatica non è appropriata, è possibile definire una sottoscrizione differente (ad esempio, utilizzando <code>IBM MQ Explorer</code>, <code>MQSC</code> o <code>PCF</code>) e indicare al programma di registrazione di utilizzare tale sottoscrizione. Ad esempio, utilizzare questo valore per suddividere in partizioni lo spazio di log in modo che un logger gestisca gli agent da A-H, un altro logger gestisca I-P e un terzo logger da Q-Z.</p> <p>Coda Se la topologia <code>IBM MQ</code> indica che la creazione di una sottoscrizione per il programma di registrazione non è conveniente, è possibile utilizzare una coda. Configurare <code>IBM MQ</code> in modo che la coda riceva i messaggi generalmente ricevuti da una sottoscrizione a <code>SYSTEM.FTE/Log/#</code> sul gestore code di coordinamento.</p>	registrazione automatica
<code>wmqfte.message.source.name</code>	Se il tipo di origine del messaggio è sottoscrizione di gestione o coda, il nome della sottoscrizione o della coda da utilizzare. Questa proprietà viene ignorata se il tipo di origine è sottoscrizione automatica.	Nessun valore predefinito
<code>wmqfte.database.credentials.file</code>	<p>Il file che contiene il nome utente e password per la connessione al database.</p> <p>Il valore di questa proprietà può contenere variabili di ambiente.</p> <p>Per ulteriori informazioni, consultare Formato file delle credenziali MFT.</p>	<p> z/OS Per informazioni sulla creazione del file delle credenziali di autenticazione, consultare Configuring MQMFTCredentials.xml on z/OS.</p> <p> ALW Per informazioni sull'ubicazione e le autorizzazioni di questo file, consultare Configuring MQMFTCredentials.xml.</p> <p> ALW Vedere anche MFT e IBM MQ autenticazione della connessione.</p>

Tabella 111. Proprietà di collegamento della modalità di collegamento per il file `logger.properties` (Continua)

Nome proprietà	Descrizione	Valore predefinito
<p><code>wmqfte.database.driver</code></p>	<p>L'ubicazione delle classi del driver JDBC per il database. Di solito, si tratta del nome file e del percorso di un file JAR.</p> <p>AIX Ad esempio, il driver di tipo 2 per Db2 sui sistemi AIX richiede il file <code>/opt/IBM/db2/V9.5/java/db2jcc.jar</code>.</p> <p>Windows Su sistemi Windows, specificare il separatore di percorso come carattere barra (<code>/</code>), ad esempio <code>C:/Program Files/IBM/SQLLIB/java/db2jcc.jar</code>.</p> <p>z/OS Su z/OS, specificare il percorso completo del file <code>db2jcc.jar</code>. Ad esempio, <code>wmqfte.database.driver=/db2/db2v10/jdbc/classes/db2jcc.jar</code>.</p> <p>z/OS Su sistemi z/OS, è necessario fare riferimento a tutti i seguenti file JAR:</p> <ul style="list-style-type: none"> • <code>db2jcc.jar</code> • <code>db2jcc_license_cisuz.jar</code> • <code>db2jcc_javax.jar</code> <p>Se il driver del proprio database è composto da più file JAR (ad esempio, Db2 V9.1 richiede un file JAR del driver e un file JAR di licenza), includere tutti questi file JAR in questa proprietà. Separare più nomi file utilizzando il separatore del percorso classi per la propria piattaforma, ossia il carattere punto e virgola (<code>;</code>) su sistemi Windows e il carattere due punti (<code>:</code>) su altre piattaforme.</p>	<p>Nessun valore predefinito</p>
<p><code>wmqfte.database.exclude_metadata</code></p>	<p>Controlla se le voci sono memorizzate nella tabella di metadati che contiene informazioni che possono essere trovate in altre tabelle all'interno dello schema del programma di registrazione database. Impostare questo valore su <code>true</code> o <code>false</code>. Queste voci di metadati non vengono più memorizzate per impostazione predefinita in quanto rappresentano una duplicazione dei dati esistenti e uno spreco di capacità di archiviazione del database. Le voci delle proprietà e le tabelle, in cui vengono visualizzati gli stessi dati, sono le seguenti:</p> <ul style="list-style-type: none"> • <code>com.ibm.wmqfte.SourceAgent TRANSFER_EVENT</code> o <code>CALL_REQUEST</code> • <code>com.ibm.wmqfte.DestinationAgent EVENTO_TRASFERIMENTO</code> • <code>com.ibm.wmqfte.MqmdUser TRANSFER_EVENT</code> o <code>CALL_REQUEST</code> • <code>com.ibm.wmqfte.OriginatingUser TRANSFER_EVENT</code> o <code>CALL_REQUEST</code> • <code>com.ibm.wmqfte.OriginatingHost TRANSFER_EVENT</code> o <code>CALL_REQUEST</code> • <code>com.ibm.wmqfte.TransferId TRANSFER</code> o <code>CALL_REQUEST</code> • <code>com.ibm.wmqfte.JobName TRANSFER</code> o <code>CALL_REQUEST</code> <p>L'impostazione del valore di tale proprietà su <code>false</code> fa sì che queste voci di metadati vengano memorizzate nella tabella dei metadati.</p>	<p>vero, true</p>

Tabella 111. Proprietà di collegamento della modalità di collegamento per il file `logger.properties` (Continua)


Nome proprietà	Descrizione	Valore predefinito
<code>wmqfte.database.host</code>	<p>Solo Db2 :</p> <p>Il nome host del server di database a cui connettersi utilizzando un driver JDBC di Tipo 4. Se viene specificato un valore per questa proprietà, è necessario specificare anche un valore per <code>wmqfte.database.port</code>. Se entrambe le proprietà non sono definite, il programma di registrazione database si connette utilizzando il driver JDBC di Tipo 2 predefinito.</p> <p>Se viene specificato un valore per questa proprietà, deve esistere un file delle credenziali per questo programma di registrazione (percorso file definito dalla proprietà <code>wmqfte.database.credentials.file</code>) e deve essere accessibile per definire il nome utente e la password per la connessione al database, anche se il database si trova sul sistema locale.</p>	Nessun valore predefinito
<code>wmqfte.database.name</code>	Il nome dell'istanza del database (o del sistema secondario quando si utilizza Db2 per z/OS) che contiene le tabelle di log Managed File Transfer .	Nessun valore predefinito
<code>wmqfte.database.type</code>	Il sistema di gestione del database in uso: Db2 o Oracle. Impostare questo valore su <code>db2</code> o <code>oracle</code> .	<code>db2</code>
<code>wmqfte.database.port</code>	<p>Solo Db2 :</p> <p>Il numero di porta del server di database a cui connettersi utilizzando un driver JDBC di Tipo 4. Se viene specificato un valore per questa proprietà, è necessario specificare anche un valore per <code>wmqfte.database.host</code>. Se entrambe le proprietà non sono definite, il programma di registrazione database si connette utilizzando il driver JDBC di Tipo 2 predefinito.</p> <p>Se viene specificato un valore per questa proprietà, deve esistere un file delle credenziali per questo programma di registrazione (percorso file definito dalla proprietà <code>wmqfte.database.credentials.file</code>) e deve essere accessibile per definire il nome utente e la password per la connessione al database, anche se il database si trova sul sistema locale.</p>	Nessun valore predefinito
<code>wmqfte.database.schema</code>	<p>Solo Db2 :</p> <p>Lo schema del database che contiene le tabelle di registrazione Managed File Transfer . Nella maggior parte dei casi il valore predefinito è appropriato, ma potrebbe essere necessario specificare un valore alternativo in base alle proprie considerazioni sul database specifiche del sito.</p>	FTELOG
<code>wmqfte.database.native.library.path</code>	<p>Il percorso che contiene le librerie native necessarie al driver del database scelto (se presente).</p> <p> Ad esempio, il driver Tipo 2 per Db2 su sistemi AIX richiede le librerie da <code>/opt/IBM/db2/V9.5/1ib32/</code>. Come alternativa a questa proprietà, è possibile impostare la proprietà di sistema <code>java.library.path</code> utilizzando altri metodi.</p>	Nessun valore predefinito
<code>wmqfte.file.logger.fileDirectory</code>	La directory in cui si trovano i file di log del programma di registrazione file.	<code>mqft/logs/coordination_dir/loggers/logger_name/logs</code>

Tabella 111. Proprietà di collegamento della modalità di collegamento per il file `logger.properties` (Continua)

Nome proprietà	Descrizione	Valore predefinito
<code>wmqfte.file.logger.fileSize</code>	La dimensione massima consentita per un file di log. Il valore della dimensione è un numero intero positivo, maggiore di zero, seguito da una delle seguenti unità: KB, MB, GB, m (minuti), h (ore), d (giorni), w (settimane). Ad esempio, <code>wmqfte.file.logger.fileSize=5MB</code> specifica una dimensione file massima di 5MB e <code>wmqfte.file.logger.fileSize=2d</code> specifica una dimensione file massima di 2 giorni di dati.	10MB
<code>wmqfte.file.logger.fileCount</code>	Il numero massimo di file di log da creare. Quando la quantità di dati supera la quantità massima che può essere memorizzata in questo numero di file, il file meno recente viene eliminato in modo che il numero di file non superi mai il valore specificato.	3

Tabella 111. Proprietà di collegamento della modalità di collegamento per il file `logger.properties` (Continua)

Nome proprietà	Descrizione	Valore predefinito
<p><code>wmqfte.file.logger.mode</code></p>	<p>La modalità logger in uso: circolare o lineare. Impostare questo valore su CIRCULARo LINEAR.</p> <p>CIRCULAR - Il programma di log di file scrive le informazioni in un file finché tale file non raggiunge la dimensione massima definita utilizzando la proprietà wmqfte.file.logger.fileSize . Quando viene raggiunta la dimensione massima, il programma di registrazione file avvia un nuovo file. Il numero massimo di file scritti in questa modalità è controllato dal valore definito utilizzando la proprietà wmqfte.file.logger.fileCount . Quando viene raggiunto questo numero massimo di file, il programma di registrazione file elimina e ricrea il primo file per utilizzarlo come file attualmente attivo. Se il valore definito nella proprietà wmqfte.file.logger.fileSize è un'unità di byte a dimensione fissa (ad esempio, KB, MB o GB), il limite superiore dello spazio su disco utilizzato in questa modalità è uguale a <code>fileSize</code> moltiplicato per <code>fileCount</code>. Se il valore definito nella proprietà wmqfte.file.logger.fileSize è un'unità di tempo (ad esempio, m, h, do w), la dimensione massima dipende dalla velocità di trasmissione dei messaggi di log nel sistema in questi periodi di tempo. La convenzione di denominazione del file di log utilizzata durante l'esecuzione in questa modalità è: <code>logger_namenumber-timestamp.log</code> dove:</p> <ul style="list-style-type: none"> • <code>logger_name</code> è il nome assegnato al logger nel comando fteCreateLogger . • <code>numero</code> è il numero del file all'interno della serie. • <code>data/ora</code> è la data/ora in cui è stato creato il file. <p>Ad esempio, <code>LOGGER1-20111216123430147.log</code></p> <p>LINEAR - Il programma di registrazione file scrive le informazioni in un file fino a quando tale file non raggiunge la dimensione massima definita utilizzando la proprietà wmqfte.file.logger.fileSize . Quando viene raggiunta la dimensione massima, il programma di registrazione file avvia un nuovo file. I file precedentemente scritti non vengono eliminati, il che consente di conservarli come record cronologico dei messaggi di log. I file non vengono eliminati durante l'esecuzione in modalità <code>lineare</code> , quindi la proprietà wmqfte.file.logger.fileCount viene ignorata perché non esiste un limite superiore al numero di file che è possibile creare. Poiché non esiste alcun limite superiore durante l'esecuzione in questa modalità, è necessario tenere traccia della quantità di spazio su disco utilizzata dai file di log per evitare che lo spazio su disco sia insufficiente. La convenzione di denominazione del file di log utilizzata durante l'esecuzione in questa modalità è: <code>logger_name-timestamp.log</code> dove:</p> <ul style="list-style-type: none"> • <code>logger_name</code> è il nome assegnato al logger nel comando fteCreateLogger . • <code>data/ora</code> è la data/ora in cui è stato creato il file. <p>Ad esempio, <code>LOGGER-20111216123430147.log</code></p>	<p>Nessun valore predefinito</p>

Tabella 111. Proprietà di collegamento della modalità di collegamento per il file `logger.properties` (Continua)

Nome proprietà	Descrizione	Valore predefinito
<code>wmqfte.max.retry.interval</code>	<p>Il tempo massimo, in secondi, tra i tentativi quando il programma di registrazione rileva un errore persistente.</p> <p>Alcune condizioni di errore (ad esempio, la perdita di connessione al database) impediscono al programma di registrazione di continuare. Quando si verifica questo tipo di condizione, il programma di registrazione esegue il rollback della transazione corrente, attende un periodo e riprova. Il tempo che il logger attende è inizialmente molto breve, in modo che gli errori transitori possano essere superati rapidamente. Tuttavia, ogni volta che il programma di registrazione ritenta, il tempo di attesa viene aumentato. Ciò impedisce che si verifichi troppo lavoro non necessario quando la condizione di errore è più lunga, ad esempio quando un database viene disattivo per manutenzione.</p> <p>Utilizzare questa proprietà per impostare un limite alla lunghezza dell'attesa in modo che un nuovo tentativo si verifichi in un periodo di tempo ragionevole in cui la condizione di errore viene risolta.</p>	600
<code>Timeout immediateShutdown</code>	<p>Il tempo, in secondi, durante il quale il programma di registrazione attende il completamento e la chiusura di tutte le operazioni in sospeso. Per impostazione predefinita, il programma di registrazione attende per 10 secondi il completamento delle operazioni. Se le operazioni non vengono completate prima del timeout, il programma di registrazione scrive il seguente messaggio di evento in <code>output0.log</code> e termina.</p> <p><code>BFGDB0082I: The logger is ending immediately.</code></p> <p>Se si specifica il valore zero, il programma di registrazione attende il completamento delle operazioni correnti.</p> <p>Il valore predefinito viene utilizzato se il valore di <code>immediateShutdownTimeout</code> è impostato su un valore inferiore a zero.</p> <p>La proprietà si applica sia al logger del database autonomo che al logger del tipo di file.</p>	10
<code>loggerCredentialsKeyFile</code>	Nome del file contenente la chiave della credenziale utilizzata durante la codifica delle credenziali.	Una proprietà stringa che non ha un valore predefinito.
<code>loggerQMGrRetryInterval</code>	L'intervallo, in secondi, tra i controlli sulla disponibilità del gestore code da parte del controller di processo del programma di registrazione.	30
<code>Conteggio maxRestart</code>	Il numero massimo di riavvii che possono verificarsi nell'intervallo di tempo specificato dal valore della proprietà <code>Intervallo maxRestart</code> . Quando questo valore viene superato, il controller di processi del programma di registrazione arresta il riavvio del programma di registrazione ed esegue invece un'operazione basata sul valore della proprietà <code>Delay maxRestart</code> .	4

Tabella 111. Proprietà di collegamento della modalità di collegamento per il file `logger.properties` (Continua)




Nome proprietà	Descrizione	Valore predefinito
Intervallo maxRestart	L'intervallo, in secondi, in cui il controller di processi del programma di registrazione misura il riavvio del programma di registrazione. Se il numero di riavvii in questo intervallo supera il valore della proprietà maxRestartCount, il controller di processo del programma di registrazione arresta il riavvio del programma di registrazione. Invece, il controller di processi del programma di registrazione esegue un'operazione basata sul valore della proprietà Ritardo maxRestart.	120
Ritardo maxRestart	Determina il funzionamento del controller di processi del programma di registrazione quando la velocità di riavvio del programma di registrazione supera il valore delle proprietà maxRestartCount e maxRestartInterval. Se si specifica un valore inferiore o uguale a zero, il controller di processo del programma di registrazione viene arrestato. Se si specifica un valore maggiore di zero, questo è il numero di secondi da attendere prima che le informazioni sulla cronologia di riavvio contenute nel controller di processi del programma di registrazione vengano reimpostate e il programma di registrazione venga riavviato.	-1
wmqfte.oracle.port	La porta utilizzata dal programma di registrazione per connettersi all'istanza Oracle . Questa porta è nota anche come listener TNS.	1521
wmqfte.oracle.host	L'host che il programma di registrazione utilizza per collegarsi all'istanza Oracle .	host locale
armeLEMTYPE	Proprietà facoltativa. Se il programma di registrazione è configurato per il riavvio da parte di ARM (Automatic Restart Manager), impostare questa proprietà sul valore del parametro ARM ELEMTYPE specificato nella politica ARM associata. Per un programma di registrazione, impostare ELEMTYPE su SYSBFGLG.	Non impostato
armeELEMENT	Proprietà facoltativa. Se il programma di registrazione è configurato per il riavvio da ARM (Automatic Restart Manager), impostare questa proprietà sul valore del parametro ARM ELEMENT specificato nella politica ARM associata. È possibile impostare il valore ELEMENT in modo che corrisponda al nome del programma di registrazione.	Non impostato
loggerQMGrAuthenticationCredentials	Il percorso del file che contiene le credenziali di connessione di MQ per la connessione al gestore code di coordinamento del programma di registrazione.	<p> Per informazioni sulla creazione del file delle credenziali di autenticazione, consultare Configuring MQMFTCredentials.xml on z/OS.</p> <p> Per informazioni sull'ubicazione e sulle autorizzazioni per questo file, consultare Configuring MQMFTCredentials.xml.</p> <p> Vedere anche MFT e IBM MQ autenticazione della connessione.</p>

Tabella 111. Proprietà di collegamento della modalità di collegamento per il file `logger.properties` (Continua)

Nome proprietà	Descrizione	Valore predefinito
traccia	<p>Proprietà facoltativa. Specifica di traccia quando il programma di registrazione deve essere eseguito con la traccia abilitata all'avvio del programma di registrazione. La specifica di traccia è un elenco separato da virgole di classi, il carattere di uguaglianza e un livello di traccia.</p> <p>Ad esempio, <code>com.ibm.wmqfte.databaseloggere</code> <code>com.ibm.wmqfte.databaselogger.operation=all</code></p> <p>È possibile specificare più specifiche di traccia in un elenco separato da due punti. Ad esempio, <code>com.ibm.wmqfte.databaselogger=moderate:</code> <code>com.ibm.wmqfte.databaselogger.operation=all</code></p>	Nessuna
traceFiles	Proprietà facoltativa. Il numero totale di file di traccia da mantenere. Questo valore si applica al controller di processi di un programma di registrazione, così come al programma di registrazione stesso.	5
traceSize	Proprietà facoltativa. La dimensione massima in MB di ciascun file di traccia, prima che la traccia si riavvolga nel file successivo. Questo valore si applica al controller di processo del programma di registrazione e al programma di registrazione stesso.	20
wmqfte.file.logger.filePermissions	<p>Proprietà facoltativa. Utilizzare per specificare quale tipo di autorizzazione è richiesto per il file di log del programma di registrazione.</p> <p>La proprietà si applica sia ai log lineari che circolari e può assumere i valori <code>UserReadWriteOnly</code> o <code>UserReadWriteAllRead</code>.</p> <p>Il valore <code>UserReadWriteOnly</code> ha l'autorità equivalente esistente di 600 e il valore <code>UserReadWriteAllRead</code> ha l'autorità equivalente di 644.</p> <p>Qualsiasi modifica nell'autorizzazione è applicabile ai file del programma di registrazione appena creati.</p> <p>Se si immette un valore per la proprietà non valido, il programma di registrazione assume il valore predefinito ed emette il messaggio BFGDB0083W nel log di output.</p>	UserReadWriteOnly

Proprietà di connessione SSL/TLS della modalità client

Le proprietà richieste per supportare la connessione in modalità client a un gestore code del programma di registrazione utilizzando SSL/TLS.

Tabella 112. Proprietà di connessione SSL/TLS della modalità client per il file `logger.properties`

Nome proprietà	Descrizione	Valore predefinito
wmqfte.queue.manager.host	Nome host o indirizzo IP del gestore code del programma di registrazione.	Nessun valore predefinito
wmqfte.queue.manager.port	Porta su cui è in ascolto il gestore code del programma di registrazione.	1414
wmqfte.queue.manager.channel	Nome del canale di connessione server sul gestore code del programma di registrazione.	SYSTEM.DEF.SVRCONN

Tabella 112. Proprietà di connessione SSL/TLS della modalità client per il file `logger.properties` (Continua)

Nome proprietà	Descrizione	Valore predefinito
<code>wmqfte.Ssl.CipherSuite</code>	<p>Specifica gli aspetti TLS del modo in cui il programma di registrazione e il gestore code del programma di registrazione scambiano i dati.</p> <p>Il valore di wmqfte.Ssl.CipherSuite è un nome CipherSuite . Il nome CipherSuite è associato al nome CipherSpec utilizzato nel canale del gestore code del logger.</p> <p>Per ulteriori informazioni, consultare CipherSuite e le associazioni di nomi CipherSpec.</p>	Nessun valore predefinito
<code>wmqfte.Ssl.PeerName</code>	<p>Specifica una struttura di nomi distinti che deve corrispondere al nome fornito dal gestore code del programma di registrazione. Il DN (distinguished name) viene utilizzato per controllare il certificato di identificazione presentato dal gestore code sulla connessione.</p>	Nessun valore predefinito
<code>wmqfte.Ssl.TrustStore</code>	<p>Specifica l'ubicazione dei certificati ritenuti attendibili dal programma di registrazione. Il valore di wmqfte.Ssl.TrustStore è un percorso file.</p> <p>Le parentesi, le virgole (,) e le barre retroverse (\) sono caratteri speciali nei comandi MFT e devono essere preceduti da un carattere barra retroversa (\).</p> <p>Windows I percorsi dei file su Windows possono essere specificati utilizzando doppie barre rovesciate (\\) come separatore o utilizzando singole barre (/).</p> <p>Si noti che il valore di questa proprietà può contenere variabili di ambiente.</p>	Nessun valore predefinito
<code>wmqfte.Ssl.TrustStoreCredentialsFile</code>	<p>Il percorso del file che contiene la credenziali wmqfte.Ssl.TrustStore .</p> <p>Si noti che il valore di questa proprietà può contenere variabili di ambiente.</p>	Nessun valore predefinito
<code>wmqfte.Ssl.TrustStoreType</code>	<p>Il tipo di keystore SSL che si desidera utilizzare. Sono supportati i keystore JKS e PKCS#12 . Il valore di questa proprietà può essere jks o pkcs12.</p>	JKS
<code>wmqfte.Ssl.KeyStore</code>	<p>Specifica la posizione della chiave privata del programma di registrazione. Il valore di wmqfte.Ssl.KeyStore è un percorso file.</p> <p>Le parentesi, le virgole (,) e le barre retroverse (\) sono caratteri speciali nei comandi MFT e devono essere preceduti da un carattere barra retroversa (\).</p> <p>Windows I percorsi dei file su Windows possono essere specificati utilizzando doppie barre rovesciate (\\) come separatore o utilizzando singole barre (/).</p> <p>Si noti che il valore di questa proprietà può contenere variabili di ambiente.</p>	Nessun valore predefinito
<code>wmqfte.Ssl.KeyStore.CredentialsFile</code>	<p>Il percorso del file che contiene la credenziali wmqfte.Ssl.KeyStore .</p> <p>Si noti che il valore di questa proprietà può contenere variabili di ambiente.</p>	Nessun valore predefinito

Tabella 112. Proprietà di connessione SSL/TLS della modalità client per il file `logger.properties` (Continua)

Nome proprietà	Descrizione	Valore predefinito
<code>wmqfte.Ssl.KeyStoreType</code>	Il tipo di keystore SSL che si desidera utilizzare. Sono supportati i keystore JKS e PKCS#12. Il valore di questa proprietà può essere <code>jks</code> o <code>pkcs12</code> .	JKS
<code>wmqfte.Ssl.FipsRequired</code>	Specifica che si desidera attivare il supporto FIPS a livello del programma di registrazione. Il valore di questa proprietà può essere <code>true</code> o <code>false</code> . Per ulteriori informazioni, consultare Supporto FIPS in MFT .	No

Concetti correlati

Proprietà SSL/TLS per MFT

Riferimenti correlati

“L'utilizzo delle variabili di ambiente nelle proprietà MFT” a pagina 169

È possibile utilizzare le variabili di ambiente nelle proprietà Managed File Transfer che rappresentano ubicazioni di file o directory. Ciò consente alle ubicazioni dei file o delle directory utilizzate durante l'esecuzione di parti del prodotto di variare in base all'ambiente corrente (ad esempio, l'utente che esegue un comando).

“Il file MFT `agent.properties`” a pagina 175

Ogni Managed File Transfer Agent ha il proprio file delle proprietà, `agent.properties`, che deve contenere le informazioni che un agente utilizza per connettersi al proprio gestore code. Il file `agent.properties` può anche contenere proprietà che modificano il comportamento dell'agent.

“File MFT `command.properties`” a pagina 203

Il file `command.properties` specifica il gestore code comandi a cui connettersi quando si immettono i comandi e le informazioni richieste da Managed File Transfer per contattare tale gestore code.

“File MFT `coordination.properties`” a pagina 199

Il file `coordination.properties` specifica i dettagli di collegamento al gestore code di coordinamento. Poiché diverse installazioni di Managed File Transfer potrebbero condividere lo stesso gestore code di coordinamento, è possibile utilizzare un link simbolico a un file `coordination.properties` comune su un'unità condivisa.

Output prodotto dalla funzione LogTransfer

Gli eventi del log di trasferimento catturano i dettagli dell'avanzamento del trasferimento dal momento in cui il trasferimento viene inoltrato fino al completamento. Le informazioni relative al trasferimento in fase di risincronizzazione vengono acquisite anche per consentire di comprendere l'avanzamento di un trasferimento.

Formato evento di trasferimento

Gli eventi di trasferimento sono in formato JSON e sono scritti nel file `transferlogN.json`, che viene creato nella directory di log dell'agent, dove N è un numero con 0 come valore predefinito. Ogni evento include i seguenti attributi comuni:

- Data e ora (in UTC)
- ID univoco

Ci sono ulteriori attributi inclusi nelle informazioni sull'evento scritte, a seconda del tipo di evento e del livello del log di trasferimento. Mentre il livello di log di trasferimento *info* scrive informazioni minime, il livello *verbose* include informazioni più dettagliate. La seguente sezione “[Eventi di esempio](#)” a pagina 218 descrive alcuni esempi di eventi di trasferimento registrati da un agente.

ID univoco

L'ID unico è incluso per aiutarti a identificare facilmente le diverse fasi man mano che un trasferimento avanza, ad esempio BFGTL0001. L'ID univoco fa parte dell'attributo **eventDescription** ed è composto da due parti:

BFGTL

Il prefisso utilizzato per tutti gli identificativi, dove BFG è il suffisso standard utilizzato in Managed File Transfer e TL indica che si tratta di un log di trasferimento.

Numero

Un numero univoco che inizia da 1. Ad esempio:

```
{
  "eventDescription": "BFGTL0001: New transfer request submitted"
}
```

Eventi di esempio

La seguente tabella descrive alcuni degli eventi come esempi delle informazioni registrate dalla funzione aggiuntiva. La seconda colonna della tabella *Livello di registrazione* indica il livello in cui viene registrato l'evento.

Importante: I seguenti attributi sono inclusi nelle informazioni sull'evento se il livello **logTransfer** è impostato su *verbose* o *moderato*:

- **sourceAgent**
- **destinationAgent**
- **threadId**

Evento	Livello di log	Descrizione
Elenco di elementi da trasferire	emissione di messaggi completi	<pre>{ "dateTime": "<Data time in UTC>", "eventDescription": "BFGTL0002I: Generated detailed transfer item list.", "destinationAgent": "<Name of destination agent>", "sourceAgent": "<Name of source agent>", "threadId": "0000001d", "totalItemsInTransfer": <Number of items in the transfer>, "transferId": "<Transfer Identifier>", "transferItemsList": [{ "source": "source item name", "destination": "destination item name" }] }</pre> <p>Example:</p> <pre>{ "dateTime": "2022-01-14T12:56:54.219Z UTC", "eventDescription": "BFGTL0002I: Generated detailed transfer item list.", "destinationAgent": "QMBAGQ", "sourceAgent": "QMBAG1", "threadId": "0000001d", "totalItems": 1, "transferId": "414d5120514d4120202020202020202063bd17610a390040", "transferItems": [{ "destination": "/results/rts/target/destFile.txt", "source": "DESTINATIONQ@QMB" }] }</pre>

Evento	Livello di log	Descrizione
Informazioni di controllo del trasferimento registrato dall'agente di origine	informazioni, moderate, verbose	<pre> { "dateTime": "<Date and time in UTC>", "transferId": "<Transfer Id>", "eventDescription": "BFGTL0008: Transfer progress information ", "progressInformation": [{ "destination": "<Destination item name>", "resultCode": "<Result of transfer>", "supplement": "<Any supplemental information.>" }], "source": "<Source item name>" }, { "sourceAgent": "<Source agent name>", "destinationAgent": "<Destination agent name>", "threadId": "Thread Id", } </pre> <p>Example:</p> <pre> { "dateTime": "2022-01-14T14:17:20.075Z UTC", "transferId": "414D5120514D41202020202020202020FD01761013B0040", "eventDescription": "BFGTL0008: Transfer progress information", "progressInformation": [{ "source": "MONITORQ@QMA", "destination": "/results/rts/target/destinationFile.txt", "resultCode": "Successful", "supplement": "" }], "sourceAgent": "QMAAG1", "destinationAgent": "QMBAG1", "threadId": "00000023", } </pre> <p>Di seguito è riportato un esempio delle informazioni registrate se il livello è info:</p> <pre> { "dateTime": "2022-01-16T03:28:26.234Z UTC", "transferId": "414D51204D4654514D2020202020202020EB89416101760040", "eventDescription": "BFGTL0008I: Transfer progress information", "progressInformation": { "warnings": 0, "failed": 1, "successful": 0, "unknown": 0 } } </pre> <p>Di seguito è riportato un esempio delle informazioni registrate se il livello è moderate:</p> <pre> { "dateTime": "2022-01-16T03:30:46.766Z UTC", "transferId": "414D51204D4654514D2020202020202020EB89416101850040", "eventDescription": "BFGTL0008I: Transfer progress information", "progressInformation": [{ "source": "FileZilla:/home/mft/*", "destination": "C:/MFT/FileZilla:/home/mft/*", "itemSize": 0, "resultCode": "Failed", "supplement": "BFGI00110E: File \" /home/mft/*\" does not exist.{}", "sourceAgent": "IMQFT01", "destinationAgent": "SRC", } } </pre>
Si è verificato un errore durante l'apertura di un elemento	informazioni	<pre> { "dateTime": "<Date and time in UTC>", "transferId": "<Transfer Id>", "eventDescription": "BFGTL0054E: Failed to open an item to transfer data.", "itemName": "Name of file or queue", "error": "<Error description>", "sourceAgent": "<Source agent name>", "destinationAgent": "<Destination agent name>", "threadId": "<Thread Id>", } </pre> <p>Example:</p> <pre> { "dateTime": "2022-01-14T14:42:24.902Z UTC", "transferId": "414D512043514D485830312020202020B0D4176101370040", "eventDescription": " BFGTL0054E: Failed to open an item to transfer data.", "itemName": "/inbound/resource.xml", "error": "File not found", "sourceAgent": "CQMHX01AG1", "destinationAgent": "HX01AG1", "threadId": "0000001c" } </pre>

Evento	Livello di log	Descrizione
Elenco dei trasferimenti da recuperare all'avvio dell'agent	emissione di messaggi completi	<pre> { "dateTime": "<Date and time in UTC>", "eventDescription": "The list of transfers being recovered as part of agent recovery process.", "agentName": "<Agent name>", "transfers": [{ "transferId": "<transfer state>" }] } Example: { "dateTime": "2022-01-14T14:42:24.902Z UTC", "eventDescription": "The list of transfers being recovered as part of agent recovery process.", "agentName": "CQMHX01AG1", "transfers": [{ "414D512043514D4858303120202020B0D4176101370040": "completeReceived" }, { "414D512043514D4858303120202020B0D4176101370050": "resynchronizing" }] } </pre>

Riferimenti correlati

“Java proprietà di sistema per MFT” a pagina 220

Un numero di proprietà del comando e dell'agent Managed File Transfer deve essere definito come proprietà di sistema Java , poiché definiscono la configurazione per la funzione precedente che non è in grado di utilizzare il meccanismo delle proprietà del comando o dell'agent.

[Agent fteCreate](#)

“L'utilizzo delle variabili di ambiente nelle proprietà MFT” a pagina 169

È possibile utilizzare le variabili di ambiente nelle proprietà Managed File Transfer che rappresentano ubicazioni di file o directory. Ciò consente alle ubicazioni dei file o delle directory utilizzate durante l'esecuzione di parti del prodotto di variare in base all'ambiente corrente (ad esempio, l'utente che esegue un comando).

Java proprietà di sistema per MFT




Un numero di proprietà del comando e dell'agent Managed File Transfer deve essere definito come proprietà di sistema Java , poiché definiscono la configurazione per la funzione precedente che non è in grado di utilizzare il meccanismo delle proprietà del comando o dell'agent.

Definire le propriet ... di sistema e altre opzioni JVM per la JVM che deve eseguire i comandi Managed File Transfer definendo la variabile di ambiente BFG_JVM_PROPERTIES. Ad esempio, per impostare la proprietà com.ibm.wmqfte.maxConsoleLineLength su una piattaforma di tipo UNIX, definire la variabile come segue:

```
export BFG_JVM_PROPERTIES="-Dcom.ibm.wmqfte.maxConsoleLineLength=132"
```

Se si sta eseguendo un agent come un servizio Windows , è possibile modificare le proprietà di sistema Java dell'agent specificando il parametro `-sj` nel comando **fteModifyAgent** .

Tabella 113. JavaProprietà di sistema

Nome proprietà	Descrizione	Valore
com.ibm.wmqfte.maxConsoleLineLength	Lunghezza massima della riga che può essere scritta sulla console. Le righe che superano questa lunghezza vengono riportate a capo. Questo valore è espresso in byte (non in caratteri).	 La lunghezza predefinita per IBM i è 132 byte.  Per AIX, Linux, and Windows, la lunghezza è illimitata.  Per z/OS, la lunghezza è illimitata.
com.ibm.wmqfte.daemon.windows.windowsServiceLogFilesm	(SoloWindows .) Specifica il numero massimo di file di log del servizio Windows da conservare. I file di log del servizio Windows vengono creati nelle directory dei log dell'agent e del programma di registrazione database se queste applicazioni sono in esecuzione come servizio Windows . I file di log del servizio Windows sono denominati con il prefisso <i>servicee</i> contengono messaggi sull'avvio e l'arresto del servizio.	5

Concetti correlati

[Opzioni di configurazione MFT su Multiplatforms](#)

[Suggerimenti e suggerimenti per l'utilizzo di MFT](#)

SHA-2 CipherSpecs e CipherSuites per MFT

Managed File Transfer supporta SHA-2 CipherSpecs e CipherSuites.

Per ulteriori informazioni su CipherSpecs e CipherSuites disponibili per connessioni tra agent e gestori code IBM MQ , consultare [TLS CipherSpecs e CipherSuites in IBM MQ classes for Java](#) e [CipherSpec CipherSpecs e CipherSuites nelle classi IBM MQ per JMS](#).

Per ulteriori informazioni sulla configurazione di CipherSpecs e CipherSuites da utilizzare con i server PBA (protocol bridge agent) e FTPS, consultare [Supporto server FTPS tramite il bridge di protocollo](#) e [Formato file delle proprietà del bridge di protocollo](#).

Se si desidera essere conformi a SP 800-131A, è necessario soddisfare i seguenti requisiti:

- È necessario utilizzare FTPS, configurato in modo appropriato; SFTP non è supportato.
- Il server remoto deve inviare solo pacchetti di crittografia compatibili con SP 800-131A-compliant .

Concetti correlati

[Proprietà SSL/TLS per MFT](#)

File di configurazione del programma di registrazione file MFT

Oltre al file `logger.properties` , un programma di registrazione file autonomo Managed File Transfer dispone anche di un file di configurazione XML nella propria directory di configurazione. Questo file di configurazione è denominato `FileLoggerFormat.xml` e definisce il formato utilizzato dal programma di registrazione file per scrivere i messaggi nel file di log. Il contenuto di questo file deve essere conforme allo schema XML definito nel file `FileLoggerFormat.xsd` .

Concetti correlati

[Formato del programma di registrazione file autonomo MFT](#)

Riferimenti correlati

[“File MFT logger.properties” a pagina 207](#)

Il programma di registrazione Managed File Transfer ha una serie di proprietà di configurazione. Specificare queste propriet ... nel file `logger.properties`, che si trova nella directory `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

[“Formato di log predefinito del programma di registrazione file autonomo MFT” a pagina 222](#)

Definizione del formato del file di log predefinito per il programma di registrazione file autonomo Managed File Transfer.

[“Formato del programma di registrazione file autonomo XSD” a pagina 227](#)

Lo schema per un formato di file autonomo.

Formato di log predefinito del programma di registrazione file autonomo MFT

Definizione del formato del file di log predefinito per il programma di registrazione file autonomo Managed File Transfer.

```
<?xml version="1.0" encoding="UTF-8"?>
<logFormatDefinition xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="1.00" xsi:noNamespaceSchemaLocation="FileLoggerFormat.xsd">
  <messageTypes>
    <callCompleted>
      <format>
        <inserts>
          <insert type="user" width="19" ignoreNull="false"/>/transaction/action/@time</insert>
          <insert type="user" width="48" ignoreNull="false"/>/transaction/@ID</insert>
          <insert type="system" width="6" ignoreNull="false">type</insert>
          <insert type="user" width="3" ignoreNull="false">/transaction/status/@resultCode</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/agent/@agent</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/agent/@QMgr</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/job/name</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/command/
@type</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/command/
@name</insert>
          <insert type="system" width="0" ignoreNull="true">callArguments</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/callResult/
@outcome</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/callResult/
result/error</insert>
        </inserts>
        <separator>;</separator>
      </format>
    </callCompleted>
    <callStarted>
      <format>
        <inserts>
          <insert type="user" width="19" ignoreNull="false"/>/transaction/action/@time</insert>
          <insert type="user" width="48" ignoreNull="false"/>/transaction/@ID</insert>
          <insert type="system" width="6" ignoreNull="false">type</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/agent/@agent</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/agent/@QMgr</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/job/name</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/command/
@type</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/command/
@name</insert>
          <insert type="system" width="0" ignoreNull="true">callArguments</insert>
        </inserts>
        <separator>;</separator>
      </format>
    </callStarted>
    <monitorAction>
      <format>
        <inserts>
          <insert type="user" width="19" ignoreNull="false"/>/monitorLog/action/@time</insert>
          <insert type="user" width="48" ignoreNull="false"/>/monitorLog/@referenceId</insert>
          <insert type="system" width="6" ignoreNull="false">type</insert>
          <insert type="user" width="3" ignoreNull="false">/monitorLog/status/@resultCode</insert>
          <insert type="user" width="0" ignoreNull="false">/monitorLog/@monitorName</insert>
          <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@agent</insert>
          <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@QMgr</insert>
          <insert type="user" width="0" ignoreNull="false">/monitorLog/action</insert>
        </inserts>
        <separator>;</separator>
      </format>
    </monitorAction>
  </messageTypes>
</logFormatDefinition>
```

```

</format>
</monitorAction>
<monitorCreate>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/monitorLog/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/monitorLog/@referenceId</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/@monitorName</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@agent</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@QMgr</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/action</insert>
    </inserts>
    <separator>;</separator>
  </format>
</monitorCreate>
<monitorFired>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/monitorLog/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/monitorLog/@referenceId</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="false">/monitorLog/status/@resultCode</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/@monitorName</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@agent</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@QMgr</insert>
      <insert type="user" width="0" ignoreNull="false">/monitorLog/action</insert>
      <insert type="user" width="48" ignoreNull="false">/monitorLog/references/taskRequest</insert>
    </inserts>
    <separator>;</separator>
  </format>
</monitorFired>
<notAuthorized>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/notAuthorized/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/notAuthorized/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="false">/notAuthorized/status/@resultCode</insert>
      <insert type="user" width="12" ignoreNull="false">/notAuthorized/action</insert>
      <insert type="user" width="12" ignoreNull="false">/notAuthorized/authority</insert>
      <insert type="user" width="0" ignoreNull="false">/notAuthorized/originator/userID</insert>
      <insert type="user" width="0" ignoreNull="false">/notAuthorized/status/supplement</insert>
    </inserts>
    <separator>;</separator>
  </format>
</notAuthorized>
<scheduleDelete>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="false">/schedulelog/status/@resultCode</insert>
      <insert type="user" width="0" ignoreNull="false">/schedulelog/sourceAgent/@agent</insert>
      <insert type="user" width="12" ignoreNull="false">/schedulelog/action</insert>
      <insert type="user" width="0" ignoreNull="false">/schedulelog/originator/userID</insert>
      <insert type="user" width="0" ignoreNull="true">/schedulelog/status/supplement</insert>
    </inserts>
    <separator>;</separator>
  </format>
</scheduleDelete>
<scheduleExpire>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="false">/schedulelog/status/@resultCode</insert>
      <insert type="user" width="0" ignoreNull="false">/schedulelog/sourceAgent/@agent</insert>
      <insert type="user" width="12" ignoreNull="false">/schedulelog/action</insert>
      <insert type="user" width="0" ignoreNull="false">/schedulelog/originator/userID</insert>
      <insert type="user" width="0" ignoreNull="true">/schedulelog/status/supplement</insert>
    </inserts>
    <separator>;</separator>
  </format>
</scheduleExpire>
<scheduleSkipped>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>

```

```

        <insert type="system" width="6" ignoreNull="false">type</insert>
        <insert type="user" width="3" ignoreNull="false">/schedulelog/status/@resultCode</insert>
        <insert type="user" width="0" ignoreNull="false">/schedulelog/sourceAgent/@agent</insert>
        <insert type="user" width="12" ignoreNull="false">/schedulelog/action</insert>
        <insert type="user" width="0" ignoreNull="false">/schedulelog/originator/userID</insert>
        <insert type="user" width="0" ignoreNull="true">/schedulelog/status/supplement</insert>
    </inserts>
    <separator>;</separator>
</format>
</scheduleSkipped>
<scheduleSubmitInfo>
    <format>
        <inserts>
            <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
            <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
            <insert type="system" width="6" ignoreNull="false">type</insert>
            <insert type="user" width="3" ignoreNull="false">/schedulelog/status/@resultCode</insert>
            <insert type="user" width="0" ignoreNull="false">/schedulelog/sourceAgent/@agent</insert>
            <insert type="user" width="12" ignoreNull="false">/schedulelog/action</insert>
            <insert type="user" width="0" ignoreNull="false">/schedulelog/originator/userID</insert>
            <insert type="user" width="0" ignoreNull="true">/schedulelog/schedule/submit</insert>
            <insert type="user" width="0" ignoreNull="true">/schedulelog/schedule/submit/@timezone</
insert>
            <insert type="user" width="3" ignoreNull="true">/schedulelog/schedule/repeat/frequency</
insert>
            <insert type="user" width="12" ignoreNull="true">/schedulelog/schedule/repeat/frequency/
@interval</insert>
            <insert type="user" width="3" ignoreNull="true">/schedulelog/schedule/repeat/expireCount</
insert>
            <insert type="user" width="0" ignoreNull="true">/schedulelog/status/supplement</insert>
        </inserts>
        <separator>;</separator>
    </format>
</scheduleSubmitInfo>
<scheduleSubmitTransfer>
    <format>
        <inserts>
            <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
            <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
            <insert type="system" width="10" ignoreNull="false">type</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/sourceAgent/@agent |
/transaction/sourceWebUser/@webGatewayAgentName |
/transaction/sourceWebGateway/@webGatewayAgentName</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/sourceAgent/@QMgr |
/transaction/sourceWebUser/@webGatewayAgentQMgr |
/transaction/sourceWebGateway/@webGatewayAgentQMgr</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/destinationAgent/@agent |
/transaction/destinationWebUser/@webGatewayAgentName |
/transaction/destinationWebGateway/@webGatewayAgentName</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/destinationAgent/@QMgr |
/transaction/destinationWebUser/@webGatewayAgentQMgr |
/transaction/destinationWebGateway/@webGatewayAgentQMgr</insert>
        </inserts>
        <separator>;</separator>
    </format>
</scheduleSubmitTransfer>
<scheduleSubmitTransferSet>
    <format>
        <inserts>
            <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
            <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
            <insert type="system" width="10" ignoreNull="false">type</insert>
            <insert type="user" width="0" ignoreNull="false">source/file | source/queue</insert>
            <insert type="user" width="5" ignoreNull="true">source/@type</insert>
            <insert type="user" width="6" ignoreNull="true">source/@disposition</insert>
            <insert type="user" width="0" ignoreNull="false">destination/file | destination/queue</
insert>
            <insert type="user" width="5" ignoreNull="true">destination/@type</insert>
            <insert type="user" width="9" ignoreNull="true">destination/@exist</insert>
        </inserts>
        <separator>;</separator>
    </format>
</scheduleSubmitTransferSet>
<transferStarted>
    <format>
        <inserts>
            <insert type="user" width="19" ignoreNull="false">/transaction/action/@time</insert>
            <insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
            <insert type="system" width="6" ignoreNull="false">type</insert>
            <insert type="user" width="3" ignoreNull="true">/transaction/status/@resultCode</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/sourceAgent/@agent |
/transaction/sourceWebUser/@webGatewayAgentName |

```



```

</format>
</transferComplete>
<transferDelete>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/transaction/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="true">/transaction/status/@resultCode</insert>
      <insert type="user" width="0" ignoreNull="false">/transaction/sourceAgent/@agent |
      /transaction/sourceWebUser/@webGatewayAgentName |
      /transaction/sourceWebGateway/@webGatewayAgentName</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/sourceAgent/@QMgr |
      /transaction/sourceWebUser/@webGatewayAgentQMgr |
      /transaction/sourceWebGateway/@webGatewayAgentQMgr</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/sourceAgent/@agentType |
      /transaction/sourceWebUser/@webGatewayAgentType |
      /transaction/sourceWebGateway/@webGatewayAgentType</insert>
      <insert type="user" width="0" ignoreNull="false">/transaction/destinationAgent/@agent |
      /transaction/destinationWebUser/@webGatewayAgentName |
      /transaction/destinationWebGateway/@webGatewayAgentName</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/destinationAgent/@QMgr |
      /transaction/destinationWebUser/@webGatewayAgentQMgr |
      /transaction/destinationWebGateway/@webGatewayAgentQMgr</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/destinationAgent/@agentType |
      /transaction/destinationWebUser/@webGatewayAgentType |
      /transaction/destinationWebGateway/@webGatewayAgentType</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/originator/userID</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/job/name</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/status/supplement</insert>
    </inserts>
    <separator>;</separator>
  </format>
</transferDelete>
<transferProgress>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/transaction/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="true">status/@resultCode</insert>
      <insert type="user" width="0" ignoreNull="false">source/file | source/queue</insert>
      <insert type="user" width="0" ignoreNull="false">source/file/@size | source/queue/@size</
insert>
      <insert type="user" width="5" ignoreNull="true">source/@type</insert>
      <insert type="user" width="6" ignoreNull="true">source/@disposition</insert>
      <insert type="user" width="0" ignoreNull="true">source/file/@alias | source/queue/@alias</
insert>
      <insert type="user" width="0" ignoreNull="true">source/file/@filesystem | source/queue/
@filesystem</insert>
      <insert type="user" width="0" ignoreNull="true">source/@correlationBoolean1</insert>
      <insert type="user" width="0" ignoreNull="true">source/@correlationNum1</insert>
      <insert type="user" width="0" ignoreNull="true">source/@correlationString1</insert>
      <insert type="user" width="0" ignoreNull="false">destination/file | destination/queue</
insert>
      <insert type="user" width="0" ignoreNull="false">destination/file/@size | destination/queue/
@size</insert>
      <insert type="user" width="5" ignoreNull="true">destination/@type</insert>
      <insert type="user" width="9" ignoreNull="true">destination/@exist</insert>
      <insert type="user" width="0" ignoreNull="true">destination/file/@alias | destination/queue/
@alias</insert>
      <insert type="user" width="0" ignoreNull="true">destination/file/@filesystem | destination/
queue/@filesystem</insert>
      <insert type="user" width="0" ignoreNull="true">destination/file/@truncateRecords</insert>
      <insert type="user" width="0" ignoreNull="true">destination/@correlationBoolean1</insert>
      <insert type="user" width="0" ignoreNull="true">destination/@correlationNum1</insert>
      <insert type="user" width="0" ignoreNull="true">destination/@correlationString1</insert>
      <insert type="user" width="0" ignoreNull="true">status/supplement</insert>
    </inserts>
    <separator>;</separator>
  </format>
</transferProgress>
</messageTypes>
</logFormatDefinition>

```

Riferimenti correlati

[Formato del programma di registrazione file autonomo MFT](#)

[“Formato del programma di registrazione file autonomo XSD” a pagina 227](#)

Lo schema per un formato di file autonomo.

Formato del programma di registrazione file autonomo XSD

Lo schema per un formato di file autonomo.

Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
@start_non_restricted_prolog@
Version: %Z% %I% %W% %E% %U% [%H% %T%]

Licensed Materials - Property of IBM

5724-H72

Copyright IBM Corp. 2011, 2024. All Rights Reserved.

US Government Users Restricted Rights - Use, duplication or
disclosure restricted by GSA ADP Schedule Contract with
IBM Corp.
@end_non_restricted_prolog@
-->

<!--
This schema defines the format of the FileLoggerFormat XML file that contains the definition
of the format to use when logging FTE log messages to a file. When an XML file that conforms
to this schema is processed by a file logger it can contain definitions for one or more
message type(s) that define how log messages of those types are output to the file log.
-->

<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema">
<xsd:include schemaLocation="fteutils.xsd"/>

<!--
Defines the logFileDefinition and version number
<logFileDefinition version="1.00" ...
  <messageTypes>
    ...
  </messageTypes>
</logFileDefinition>
-->
<xsd:element name="logFileDefinition">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="messageTypes" type="messageTypesType" maxOccurs="1" minOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute name="version" type="versionType" use="required"/>
  </xsd:complexType>
</xsd:element>

<!--
Defines the set of accepted message types. The definition of individual message types
is optional. If a particular types element is present but empty then no line will be
output for messages of that type. If a particular types element is not present then
the default format will be used to format messages of that type.
-->
<xsd:complexType name="messageTypesType">
  <xsd:sequence>
    <xsd:element name="callCompleted" type="messageType" maxOccurs="1"
minOccurs="0"/>
    <xsd:element name="callStarted" type="messageType" maxOccurs="1"
minOccurs="0"/>
    <xsd:element name="monitorAction" type="messageType" maxOccurs="1"
minOccurs="0"/>
    <xsd:element name="monitorCreate" type="messageType" maxOccurs="1"
minOccurs="0"/>
    <xsd:element name="monitorFired" type="messageType" maxOccurs="1"
minOccurs="0"/>
    <xsd:element name="notAuthorized" type="messageType" maxOccurs="1"
minOccurs="0"/>
    <xsd:element name="scheduleDelete" type="messageType" maxOccurs="1"
minOccurs="0"/>
    <xsd:element name="scheduleExpire" type="messageType" maxOccurs="1"
minOccurs="0"/>
    <xsd:element name="scheduleSkipped" type="messageType" maxOccurs="1"
minOccurs="0"/>
    <xsd:element name="scheduleSubmitInfo" type="messageType" maxOccurs="1"
minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

```

```

        <xsd:element name="scheduleSubmitTransfer" type="messageType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="scheduleSubmitTransferSet" type="messageType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="transferStarted" type="messageType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="transferCancelled" type="messageType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="transferComplete" type="messageType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="transferDelete" type="messageType" maxOccurs="1"
minOccurs="0"/>
        <xsd:element name="transferProgress" type="messageType" maxOccurs="1"
minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>

<!--
    Defines the content of a message type definition e.g.

    <callStarted>
    <format>
    ...
    </format>
    <callStarted>
-->
<xsd:complexType name="messageType">
    <xsd:sequence>
        <xsd:element name="format" type="messageFormatType" maxOccurs="1" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>

<!--
    Defines the content of a message format definition e.g.

    <format>
    <inserts>
    ...
    </inserts>
    <separator>;</separator>
    </format>
-->
<xsd:complexType name="messageFormatType">
    <xsd:sequence>
        <xsd:element name="inserts" type="insertsType" maxOccurs="1" minOccurs="1"/>
        <xsd:element name="separator" type="scheduleType" maxOccurs="1" minOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>

<!--
    Defines the content of the inserts element e.g.

    <inserts>
    <insert ...>
    <insert ...>
    ...
    </inserts>
-->
<xsd:complexType name="insertsType">
    <xsd:sequence>
        <xsd:element name="insert" type="insertType" maxOccurs="unbounded" minOccurs="1"/>
    </xsd:sequence>
</xsd:complexType>

<!--
    Defines the content of an insert definition e.g.

    <insert type="user" width="0" ignoreNull="true">/transaction/@ID</insert>
-->
<xsd:complexType name="insertType">
    <xsd:attribute name="type" type="insertTypeType" use="required"/>
    <xsd:attribute name="width" type="xsd:nonNegativeInteger" use="required"/>
    <xsd:attribute name="ignoreNull" type="xsd:boolean" use="required"/>
</xsd:complexType>

<!--
    Defines the accepted choices for the insert type attribute.
-->
<xsd:simpleType name="insertTypeType">
    <xsd:restriction base="xsd:token">
        <xsd:enumeration value="user"/>
        <xsd:enumeration value="system"/>
    </xsd:restriction>
</xsd:simpleType>

```

```
</xsd:restriction>  
</xsd:simpleType>
```

```
</xsd:schema>
```

Riferimenti correlati

Formato del programma di registrazione file autonomo MFT

“Formato di log predefinito del programma di registrazione file autonomo MFT” a pagina 222

Definizione del formato del file di log predefinito per il programma di registrazione file autonomo Managed File Transfer .

z/OS La libreria SCSQFCMD

La libreria SCSQFCMD fornita da IBM MQ Managed File Transfer for z/OS contiene membri che fungono da modelli per i lavori che possono essere utilizzati per creare una configurazione Managed File Transfer e per creare e gestire un agent o un programma di registrazione.

Il contenuto della libreria viene visualizzato nella seguente tabella.

Membro	Descrizione
BFGCOPY	Lavoro utilizzato per creare una copia della libreria SCSQFCMD
BFGCUSTM	Lavoro utilizzato per personalizzare una copia della libreria per un agent o un logger
BFGXCROB	Modello di esempio fteObfuscate .
BFGXLGCR	Modello fteCreateLogger .
BFGXMNCR	Modello di esempio fteCreateMonitor .
BFGXMNDE	Modello di esempio fteDeleteMonitor .
BFGXPRAN	fteAnt Modello di esempio
BFGXSTDE	fteDeleteScheduledTransfer Modello di esempio
BFGXTMCR	fteCreateTemplate Modello di esempio
BFGXTMDE	fteDeleteTemplate Modello di esempio
BFGXTRCA	fteCancelTransfer Modello di esempio
BFGXTRCR	fteCreateTransfer Modello di esempio
GGGAGB	Modello per una procedura di attività avviata per avviare un agente
BFGGGGGST	Modello per una procedura di attività avviata per avviare un programma di registrazione
BFGZAGCL	fteCleanAgent Modello di esempio
BFGZAGCR	fteCreateAgent Modello di esempio
BFGZAGDE	fteDeleteAgent Modello di esempio
BFGZAGLG	fteSetAgentLogLevel Modello di esempio
BFGZAGI	fteListAgents Modello di esempio
BFGZAGPI	ftePingAgent Modello di esempio
BFGZAGSH	fteShowAgentDetails Modello di esempio
BFGZAGSP	fteStopAgent Modello di esempio
BFGZAGST	fteStartAgent Modello di esempio

Membro	Descrizione
BFGZAGC	fteSetAgentTraceLevel Modello di esempio
BFGZCFR	fteSetupCoordination Modello di esempio
BFGZCFF	fteChangeDefaultConfigurationOptions Modello di esempio
BFGZCMCR	fteSetupCommands Modello di esempio
BFGZCMD	Modello per lo script REXX utilizzato da altri membri nel dataset
BFGZLGDE	fteDeleteLogger Modello di esempio
BFGZLGSH	fteShowLoggerDetails Modello di esempio
BFGZLGSP	fteStopLogger Modello di esempio
BFGZLGST	fteStartLogger Modello di esempio
BFGZLGC	fteSetLoggerTraceLevel Modello di esempio
BFGZMNL	fteListMonitors Modello di esempio
IDPZGBF	fteSetProductId Modello di esempio
BFGZPROF	Modello per lo script della shell utilizzato da altri membri nel dataset
BFGZPRSH	fteDisplayVersion Modello di esempio
BFGZRAS	fteRas Modello di esempio
ZSTLI BF	fteListScheduledTransfers Modello di esempio
BFGZTMLI	fteListTemplates Modello di esempio

Per i dettagli su come la libreria SCSQFCMD viene utilizzata per generare una nuova libreria per la creazione di una configurazione Managed File Transfer e per la creazione e la gestione di un agent o di un logger, consultare [Configurazione di Managed File Transfer for z/OS](#).

Riferimenti correlati

“L'utilizzo delle variabili di ambiente nelle proprietà MFT” a pagina 169

È possibile utilizzare le variabili di ambiente nelle proprietà Managed File Transfer che rappresentano ubicazioni di file o directory. Ciò consente alle ubicazioni dei file o delle directory utilizzate durante l'esecuzione di parti del prodotto di variare in base all'ambiente corrente (ad esempio, l'utente che esegue un comando).

SYSTEM.FTE FTE

Il SISTEMA SYSTEM.FTE FTE è un argomento sul gestore code di coordinamento che Managed File Transfer utilizza per registrare i trasferimenti e memorizzare le informazioni su agent, monitoraggi, pianificazioni e modelli.

Struttura argomento

```
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/nome_agent

Questo argomento contiene una pubblicazione conservata che descrive un agent nella rete Managed File Transfer e le relative proprietà. Il messaggio su questo argomento viene aggiornato periodicamente con lo stato dell'agente. Per ulteriori informazioni, consultare [MFT agent status message format](#).

SYSTEM.FTE/monitors/nome_agent

Questo argomento contiene pubblicazioni conservate che descrivono i monitoraggi delle risorse associati all'agent *nome_agent*. L'XML della pubblicazione conservata è conforme allo schema `MonitorList.xsd`. Per ulteriori informazioni, consultare [MFT monitor list message format](#).

SYSTEM.FTE/Scheduler/nome_agent

Questo argomento contiene una pubblicazione conservata che descrive tutte le pianificazioni attive associate all'agent *nome_agent*. L'XML della pubblicazione conservata è conforme allo schema `ScheduleList.xsd`. Per ulteriori informazioni, consultare [Formato del messaggio dell'elenco di pianificazioni MFT](#).

SYSTEM.FTE/Templates

Questo argomento contiene pubblicazioni conservate che descrivono tutti i modelli definiti nella topologia Managed File Transfer .

- La pubblicazione associata a ciascun modello viene pubblicata in un argomento secondario denominato `SYSTEM.FTE/Templates/template_ID`.

Per un esempio del contenuto di questa pubblicazione conservata, consultare [Messaggio XML modello di esempio MFT](#).

SYSTEM.FTE/Transfers/nome_agent

Questo argomento contiene pubblicazioni che descrivono lo stato dei trasferimenti che hanno origine nell'agent *nome_agent*. Le pubblicazioni associate a ciascun trasferimento vengono pubblicate in un argomento secondario denominato `SYSTEM.FTE/Transfers/agent_name/transfer_ID`. Queste pubblicazioni vengono utilizzate dal plugin IBM MQ Explorer per fornire informazioni sull'avanzamento dei singoli trasferimenti. L'XML della pubblicazione è conforme allo schema `TransferStatus.xsd`. Per ulteriori informazioni, vedere [Formato del messaggio di stato del trasferimento file](#).

SYSTEM.FTE/Log/nome_agent

Questo argomento contiene pubblicazioni che registrano informazioni su trasferimenti, monitoraggi e pianificazioni che hanno origine nell'agent *nome_agent*. Queste pubblicazioni possono essere registrate dal programma di registrazione database per fornire i record di verifica degli eventi che si verificano nella rete Managed File Transfer .

- Le pubblicazioni associate a ciascun trasferimento vengono pubblicate in un argomento secondario con il nome `SYSTEM.FTE/Log/agent_name/transfer_ID` e l'XML della pubblicazione è conforme allo schema `TransferLog.xsd`. Per ulteriori informazioni, consultare [Formati dei messaggi del log di trasferimento file](#).
- Le pubblicazioni associate a ogni trasferimento pianificato vengono pubblicate in un argomento secondario con il nome `SYSTEM.FTE/Log/agent_name/schedule_ID` e l'XML della pubblicazione è conforme allo schema `ScheduleLog.xsd`. Per ulteriori informazioni, consultare [Formati dei messaggi di log di trasferimento file pianificati](#).
- Le pubblicazioni associate a ciascun controllo vengono pubblicate in un argomento secondario denominato `SYSTEM.FTE/Log/agent_name/Monitors/monitor_name/monitor_ID` e l'XML della pubblicazione è conforme allo schema `MonitorLog.xsd`. Per ulteriori informazioni, consultare [MFT monitor log message format](#).

MFT Impostazioni coda agent

Gli script di comandi MQSC generati dal comando **fteCreateAgent** creano le code agent con i parametri impostati sui valori seguenti. Se non si utilizzano gli script MQSC forniti per creare le code, ma si creano le code manualmente, assicurarsi di impostare i seguenti parametri sui valori forniti.

Code operazioni agent

Le code operazioni dell'agente hanno i nomi seguenti:

- SYSTEM.FTE.COMMAND.*nome_agent*
- SYSTEM.FTE.DATA.*nome_agent*
- SYSTEM.FTE.EVENT.*nome_agent*
- SYSTEM.FTE.REPLY.*nome_agent*
- SYSTEM.FTE.STATE.*nome_agent*

Parametro	Valore (se applicabile)
DEFPRTY	0
DEFSOPT	CONDIVISO
GET	Abilitato
MAXDEPTH	5000
MAXMSGL	4194304
MSGDLVSQ	PRIORITY
PUT	Abilitato
RETINTVL	999999999
SHARE	
Nessun trigger	
USAGE	NORMALE
SOSTITUISCI	

Code di autorizzazione agent

Le code di autorizzazione dell'agente hanno il seguente nome:

- SYSTEM.FTE.AUTHADM1.*nome_agent*
- SYSTEM.FTE.AUTHAGT1.*nome_agent*
- SYSTEM.FTE.AUTHMON1.*nome_agent*
- SYSTEM.FTE.AUTHOPS1.*nome_agent*
- SYSTEM.FTE.AUTHSCH1.*nome_agent*
- SYSTEM.FTE.AUTHTRN1.*nome_agent*

Parametro	Valore (se applicabile)
DEFPRTY	0
DEFSOPT	CONDIVISO
GET	Abilitato

<i>Tabella 115. Parametri della coda di autorizzazione agent (Continua)</i>	
Parametro	Valore (se applicabile)
MAXDEPTH	0
MAXMSGL	0
MSGDLVSQ	PRIORITY
PUT	Abilitato
RETINTVL	999999999
SHARE	
Nessun trigger	
USAGE	NORMALE
SOSTITUISCI	

Riferimenti correlati

[fteCreateAgent \(crea un agent MFT\)](#)

Code di sistema MFT e argomento di sistema

Managed File Transfer ha un certo numero di code di sistema e un argomento di sistema che sono solo per uso interno.

Tutte le code con un nome che inizia con SYSTEM.FTE sono code di sistema interne per Managed File Transfer (MFT). Non eliminare queste code, in quanto ciò impedisce a IBM MQ MFT di funzionare correttamente. La [Tabella 116 a pagina 233](#) mostra quale tipo di messaggio è presente su ciascuna coda:

<i>Tabella 116. Nomi coda, tipo e utilizzo</i>		
Nome coda	Tipo coda	Utilizzo
SYSTEM.FTE.AUTHAGT1.agent_name	Autorizzazione	Coda per la configurazione dell'autorità per l'invio e la ricezione di richieste di trasferimento.
SYSTEM.FTE.AUTHTRN1.agent_name	Autorizzazione	Coda per configurare l'autorizzazione per avviare e annullare i trasferimenti gestiti. Anche per avviare chiamate gestite.
SYSTEM.FTE.AUTHMON1.agent_name	Autorizzazione	Coda per la configurazione dell'autorizzazione per consentire a un utente di creare o eliminare i monitoraggi risorse creati dallo stesso utente.
SYSTEM.FTE.AUTHOPS1.agent_name	Autorizzazione	Coda per la configurazione dell'autorità per eliminare i monitoraggi risorse e i trasferimenti pianificati creati da un altro utente.

Tabella 116. Nomi coda, tipo e utilizzo (Continua)

Nome coda	Tipo coda	Utilizzo
SYSTEM.FTE.AUTHSCH1.agent_name	Autorizzazione	Coda per la configurazione dell'autorità per creare o eliminare i trasferimenti pianificati creati dallo stesso utente.
SYSTEM.FTE.AUTHADM1.agent_name	Autorizzazione	Coda per la configurazione dell'autorità per arrestare l'agente, utilizzando l'opzione -m sul comando fteStopAgent .
SYSTEM.FTE.COMMAND.agent_name	Operazione	Coda per l'invio di richieste di comandi ad un agente.
SYSTEM.FTE.DATA.agent_name	Operazione	Coda utilizzata da un agent di destinazione per contenere i dati inviati da un agent di origine.
SYSTEM.FTE.REPLY.agent_name	Operazione	Coda per ricevere risposte da un agente di destinazione.
SYSTEM.FTE.STATE.agent_name	Operazione	Coda per il congelamento dello stato di una richiesta di trasferimento.
SYSTEM.FTE.EVENT.agent_name	Operazione	Coda per il congelamento della cronologia del controllo risorse.
SYSTEM.FTE.HA.agent_name	Operazione	Coda utilizzata come blocco da istanze agent ad alta disponibilità.

Se un agent partecipa a trasferimenti da messaggio a file o da file a messaggio, potrebbe essere necessario modificare la definizione della coda `SYSTEM.FTE.STATE.agent_name` per consentire l'esecuzione di tali trasferimenti gestiti. Per ulteriori informazioni, consultare [Guida per l'impostazione degli attributi MQ e le proprietà MFT associate alla dimensione del messaggio](#).



Attenzione: Non modificare le definizioni delle altre code di sistema.

Inoltre, non modificare o eliminare `SYSTEM.FTE.FTE` poiché questo argomento è solo per uso interno.

Code temporanee

Managed File Transfer crea code temporanee per diversi scopi. Il nome di ogni coda inizia con `WMQFTE`, per impostazione predefinita. (Il periodo fa parte del prefisso predefinito.) Se si desidera modificare questo prefisso, è possibile utilizzare la proprietà **dynamicQueuePrefix** nel file `command.properties` o nel file `coordination.properties` o in entrambi. La proprietà nel file `command.properties` viene utilizzata per impostare il prefisso delle code temporanee create per le risposte ai comandi che richiedono una risposta dall'agente. La proprietà nel file `coordination.properties` viene utilizzato per impostare il prefisso delle code temporanee create per altri scopi; ad esempio, l'`FTE.WMQFTE.FTE.TIMECHK.QUEUE`, dove `WMQFTE` è il valore definito dalla proprietà **dynamicQueuePrefix**.

Riferimenti correlati

[Limitazione delle autorizzazioni utente sulle azioni dell'agent MFT](#)

Convenzioni di denominazione degli oggetti MFT


Utilizzare le convenzioni di denominazione riportate di seguito per gli oggetti Managed File Transfer :

- Nomi agente e programma di registrazione:
 - Può avere una lunghezza massima di 28 caratteri e non è sensibile al maiuscolo / minuscolo.
 - Le lettere minuscole o miste vengono convertite in maiuscole
 - Deve essere conforme alle convenzioni di denominazione degli oggetti IBM MQ standard.

Queste convenzioni sono dettagliate come segue: [Regole per la denominazione di oggetti IBM MQ](#).
- Oltre alle convenzioni di denominazione degli oggetti IBM MQ , la:
 - Il carattere barra (/) non può essere utilizzato nei nomi degli agenti o nei nomi dei logger
 - Il carattere percentuale (%) non può essere utilizzato nei nomi agent o nei nomi del programma di registrazione.
- i nomi delle proprietà nei file delle proprietà sono sensibili al maiuscolo / minuscolo.
- I nomi dei gestori code sono sensibili al maiuscolo/minuscolo.
- I nomi file sono sensibili al maiuscolo / minuscolo per alcune piattaforme.
- Nomi modello di trasferimento e monitoraggio risorse:
 - Non sono sensibili al maiuscolo / minuscolo
 - Le lettere minuscole o miste vengono convertite in maiuscole
 - Non deve contenere caratteri asterisco (*), percentuale (%) o punto interrogativo (?)
- I nomi server di file del protocollo devono:
 - Deve avere una lunghezza minima di 2 caratteri, ma non esiste un limite massimo di lunghezza
 - Non sono sensibili al maiuscolo / minuscolo
 - Deve essere conforme alle convenzioni di denominazione degli oggetti IBM MQ standard.

Queste convenzioni sono dettagliate come segue: [Regole per la denominazione di oggetti IBM MQ](#).


File nell'IFS (integrated file system) IBM i

 I nomi file in IFS non possono contenere i seguenti caratteri:

- Barra retroversa (\)
- Barra (/)
- Due punti (:)
- Asterisco (*)
- Punto interrogativo (?)
- Virgolette (")
- Simbolo minore di (<)
- Simbolo maggiore di (>)
- Barra verticale (|)

Se si tenta di trasferire i file con nomi che contengono uno di questi caratteri a un IFS IBM i , il trasferimento di questi file non riesce.

Nomi dataset

 I dataset hanno restrizioni di denominazione, che influiscono sulla lunghezza massima del nome e sui caratteri disponibili che possono essere utilizzati per i nomi dei dataset. I nomi dei membri del dataset PDS possono avere una lunghezza massima di otto caratteri e non possono contenere il carattere punto (.). Quando si esegue il trasferimento a un dataset, è necessario specificare esplicitamente il nome,

il che significa che queste restrizioni di denominazione non causano un problema. Ma quando si effettua il trasferimento da file a membri PDS, il percorso file potrebbe non essere associato a un nome membro PDS. Quando si esegue il trasferimento a un dataset PDS, ogni file di origine diventa membro PDS e ogni nome membro viene generato dal nome dell'origine.

I nomi membro PDS sono z/OS nomi non qualificati e sono definiti dalla seguente espressione regolare:

```
[a-zA-Z$#@][a-zA-Z0-9$#@]{0-7}
```

Il seguente schema viene utilizzato per convertire un dataset di origine o un nome file di origine in un nome membro PDS valido. Le considerazioni vengono applicate nell'ordine elencato:

1. Vengono utilizzati solo i caratteri nel nome dopo l'ultima barra (/), l'ultima barra rovesciata (\) o l'ultimo carattere due punti (:). Ciò significa che viene utilizzata solo la parte del nome di un percorso file.
2. Per i file di origine (non i data set o i membri PDS), i caratteri che seguono e includono l'ultimo carattere punto (.), vengono ignorati.
3. Per qualsiasi nome più lungo di otto caratteri, vengono utilizzati solo gli ultimi otto caratteri.
4. I caratteri punto vengono sostituiti con i caratteri chiocciola (@).
5. I caratteri non validi vengono sostituiti con i caratteri chiocciola (@).
6. Se la conversione non produce caratteri, il nome del membro PDS è @.

Messaggi di stato dell'agente MFT

Gli agenti ad alta disponibilità pubblicano le informazioni sullo stato in formato XML.

XML di esempio che mostra le informazioni su tre istanze in standby

```
<?xml version="1.0" encoding="UTF-8"?>
<AgentStandbyStatus version="6.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="AgentStandbyStatus.xsd">
  <instance host="9.122.123.124" agentVersion="9.1.4.0" />
  <instance host="agenthost.ibm.com" agentVersion="9.1.4.0" />
  <instance host="10.11.12.14" agentVersion="9.1.4.0" />
</AgentStandby>
```

Pubblicazione dello stato dell'agent con XML di stato standby integrato.

Lo stato standby XML viene visualizzato in grassetto.

```
<?xml version="1.0" encoding="UTF-8"?>
<properties version="1.0">
  <entry key="SourceTransferStates"/>
  <entry key="queueManagerPort">1414</entry>
  <entry key="agentStandbyInstances">&lt;?xml version="1.0" encoding="UTF-8"?&gt;&lt;AgentStandbyStatus
version="6.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="AgentStandbyStatus.xsd"&gt;&lt;Instances&gt;&lt;instance
host="9.122.123.124"
  agentVersion="9.1.4.0" /&gt;&lt;instance host="agenthost.ibm.com" agentVersion="9.1.4.0" /
&gt;&lt;instance host="10.11.12.14"
  agentVersion="9.1.4.0" /&gt;&lt;/Instances&gt;&lt;/AgentStandbyStatus&gt;</entry>
  <entry key="agentType">STANDARD</entry>
  <entry key="agentDeclaredHostName">MFTHA1</entry>
  <entry key="agentDescription"/>
  <entry key="maxQueuedTransfers">1000</entry>
  <entry key="agentTimeZone">America/Los_Angeles</entry>
  <entry key="agentOsName">Windows Server 2012 R2</entry>
  <entry key="PublishTimeUTC">2019-05-22T06:02:50Z</entry>
  <entry key="queueManagerHost">localhost</entry>
  <entry key="AgentStartTimeUTC">2019-05-22T04:13:02Z</entry>
  <entry key="agentTraceLevel">&lt;?xml version="1.0" encoding="UTF-8"?&gt;&lt;
agentTraceStatus version="6.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="AgentTraceStatus.xsd"&gt;&lt;trace
level="all"&gt;com.ibm.wmqfte&lt;/trace&gt;&lt;/agentTraceStatus&gt;</entry>
```

```
<entry key="DestinationTransferStates"/>
<entry key="queueManager">MFTHAQM</entry>
<entry key="agentProductVersion">9.1.4.0</entry>
<entry key="AgentStatusPublishRate">300</entry>
<entry key="maxSourceTransfers">25</entry>
<entry key="AgentStatus">STARTED</entry>
<entry key="maxDestinationTransfers">25</entry>
<entry key="agentName">SRC</entry>
<entry key="CommandTimeUTC">2019-05-22T06:02:50Z</entry>
<entry key="queueManagerChannel">MFT_HA_CHN</entry>
<entry key="agentInterfaceVersion">6.00</entry>
<entry key="agentVersion">p914-L191119</entry>
</properties>
```

Riferimenti correlati

[Agent fteCreate](#)

[GET agent](#)

IBM MQ Internet Pass-Thru Riferimento di configurazione

IBM MQ Internet Pass-Thru (MQIPT) usa un file di configurazione denominato `mqipt.conf` per definire gli instradamenti e controllare le azioni del server MQIPT. Da IBM MQ 9.2, è possibile specificare anche le proprietà di configurazione per il comando `mqiptAdmin` in un file delle proprietà.

Il file di configurazione MQIPT

Il file di configurazione MQIPT comprende una serie di sezioni. C'è una sezione `[global]` e una sezione `[route]` aggiuntiva per ogni instradamento attraverso MQIPT che è stato definito.

Ogni sezione contiene coppie di proprietà nome / valore. Alcune proprietà possono essere visualizzate solo nella sezione `[global]`, altre solo nelle sezioni `[route]` e alcune possono essere visualizzate sia nelle sezioni `[route]` che `[global]`. Se una proprietà viene visualizzata sia nelle sezioni di instradamento che `[global]`, il valore della proprietà nella sezione `[route]` sovrascrive il valore globale, ma solo per l'instradamento in questione. In tal modo, la sezione `[global]` può essere utilizzata per stabilire i valori predefiniti da utilizzare per quelle proprietà che non sono impostate nelle singole sezioni `[route]`.

La sezione `[global]` inizia con una riga contenente i caratteri `[global]` e termina quando inizia la prima sezione `[route]`. La sezione `[global]` deve precedere tutte le sezioni `[route]` nel file.

Ogni sezione `[route]` inizia con una riga contenente i caratteri `[route]` e termina quando inizia la sezione `[route]` successiva o quando viene raggiunta la fine del file di configurazione.

Qualsiasi nome di proprietà non riconosciuto viene ignorato. Se una proprietà in una sezione `[route]` ha un nome riconosciuto ma ha un valore non valido (ad esempio `MinConnectionThreads=x` o `HTTP=unsure`), tale instradamento è disabilitato (ovvero, non è in ascolto di alcuna connessione in entrata).



Attenzione: Il limite massimo per il numero di instradamenti che possono essere aggiunti nel file `mqipt.conf` è 100.

Valori non validi per le proprietà nella sezione `[global]` potrebbero impedire l'avvio di MQIPT del server dei comandi. Se il server dei comandi non viene avviato, MQIPT non ascolta i comandi di gestione inviati dal comando `mqiptAdmin` alla porta dei comandi interessata. Se le proprietà con valori non validi nella sezione `[global]` sono presenti quando MQIPT viene aggiornato, viene emesso un messaggio di avviso e il valore effettivo della proprietà rimane invariato. Ciò impedisce ai valori di proprietà non validi di causare l'arresto di un'istanza attiva di MQIPT quando viene aggiornata.

Quando una proprietà viene elencata come se assumesse i valori `true` o `false`, qualsiasi combinazione di caratteri maiuscoli e minuscoli può essere utilizzata nel valore della proprietà.

È possibile modificare il valore di una proprietà modificando il file `mqipt.conf`. Per applicare le modifiche, aggiornare MQIPT utilizzando il comando `mqiptAdmin` con la parola chiave **-refresh**.

Per includere commenti nel file di configurazione, iniziare una riga con un carattere `"#"`.

Le modifiche a determinate proprietà causano il riavvio di un instradamento solo se altre proprietà sono già abilitate. Ad esempio, qualsiasi modifica alle proprietà HTTP ha effetto solo se è abilitata anche la proprietà **HTTP**.

Quando un instradamento viene riavviato, le connessioni esistenti vengono terminate. Per sovrascrivere questo comportamento, impostare la proprietà **RouteRestart** su `false`. Ciò impedisce il riavvio dell'instradamento, consentendo alle connessioni esistenti di rimanere attive finché la proprietà **RouteRestart** non viene riabilitata.

Per informazioni su come configurare alcune configurazioni semplici, vedi [Introduzione a MQIPT](#). Per una configurazione di esempio, consultare il file `mqiptSample.conf` nella directory di installazione MQIPT.

Il file delle proprietà mqiptAdmin

Le proprietà di configurazione per il comando **mqiptAdmin** possono essere specificate in un file delle proprietà separato. Queste proprietà di configurazione sono necessarie quando **mqiptAdmin** si connette alla porta comandi TLS MQIPT.

Per l'elenco delle proprietà che può essere specificato nel file delle proprietà **mqiptAdmin**, consultare ["Proprietà di mqiptAdmin" a pagina 267](#). I nomi proprietà sono sensibili al maiuscolo e al minuscolo. Tutte le proprietà non riconosciute vengono ignorate.

I commenti possono essere inseriti nel file delle proprietà iniziando una riga con un carattere "#".

Riepilogo delle proprietà MQIPT

Questa tabella mostra un riepilogo di proprietà di configurazione MQIPT e include le seguenti informazioni:

- Un elenco alfabetico di proprietà MQIPT con collegamenti a ulteriori informazioni nella sezione `[route]` o nella sezione `[global]` se la sezione `[route]` non si applica.
- La proprietà che deve essere impostata su `true` affinché un valore abbia effetto.
- Se la proprietà si applica alla sezione `[global]`, alla sezione `[route]` o a entrambe.
- I valori predefiniti utilizzati se una proprietà non è presente sia nella sezione `[route]` che nella sezione `[global]`. quando si specificano i valori `true` e `false`, è possibile utilizzare qualsiasi combinazione di caratteri maiuscoli e minuscoli.



Nome della proprietà	Proprietà da impostare true	Globale	Instradamento	Valore predefinito
AccessPW		sì	no	vuoto
Attivo		sì	sì	vero, true
V9.4.0 V9.4.0 "[MQ 9.4.0 giu 2024][MQ 9.4.0 giu 2024]AllowedProtocols" a pagina 247		sì	sì	mq
ClientAccess		sì	sì	No
CommandPort		sì	no	vuoto
CommandPortListenerAddress		sì	no	vuoto
ConnectionLog		sì	no	vero, true
Destinazione		no	sì	vuoto
DestinationPort		no	sì	1414
"Funzioni EnableAdvanced" a pagina 244		sì	no	No

Nome della proprietà	Proprietà da impostare true	Globale	Instradamento	Valore predefinito
HTTP		sì	sì	No
V9.4.0 V9.4.0 "[MQ 9.4.0 giu 2024][MQ 9.4.0 giu 2024]HTTPConnectionTimeout" a pagina 248		sì	sì	5000
Proxy HTTP	HTTP	sì	sì	vuoto
HTTPProxyPort	HTTP	sì	sì	8080
HTTPS	HTTP	sì	sì	No
Server HTTP	HTTP	sì	sì	vuoto
HTTPServerPort	HTTP	sì	sì	vuoto
IdleTimeout		sì	sì	0
IgnoreExpiredCRL		sì	sì	No
LDAP		sì	sì	No
LDAPIgnoreErrors	LDAP	sì	sì	No
LDAPCacheTimeout	LDAP	sì	sì	24
LDAPServer1	LDAP	sì	sì	vuoto
LDAPServer1Port	LDAP	sì	sì	389
LDAPServer1Userid	LDAP	sì	sì	vuoto
LDAPServer1Password	LDAP	sì	sì	vuoto
LDAPServer1Timeout	LDAP	sì	sì	0
LDAPServer2	LDAP	sì	sì	vuoto
LDAPServer2Port	LDAP	sì	sì	389
LDAPServer2Userid	LDAP	sì	sì	vuoto
LDAPServer2Password	LDAP	sì	sì	vuoto
LDAPServer2Timeout	LDAP	sì	sì	0
ListenerAddress		sì	sì	vuoto
ListenerPort		no	sì	vuoto
LocalAddress		sì	sì	vuoto
LocalAdmin		sì	no	vero, true
MaxConnectionThread		sì	sì	100
MaxLogFileSize		sì	no	50
MinConnectionThread		sì	sì	5
Nome		no	sì	vuoto
OutgoingPort		no	sì	0
V9.4.0 PasswordProtection		sì	sì	obbligatorio


Nome della proprietà	Proprietà da impostare true	Globale	Instradamento	Valore predefinito
QMGrAccess		sì	sì	vero, true
AutenticazioneRemoteCommand		sì	no	Nessuno
RemoteShutdown		sì	no	No
RouteRestart		sì	sì	vero, true
SecurityExit		sì	sì	No
SecurityExitNome	SecurityExit	sì	sì	vuoto
SecurityExitSecurityExit	SecurityExit	sì	sì	<i>mqi</i> pt_home \\exits
SecurityExitTimeout	SecurityExit	sì	sì	30
SecurityManager (Nota 3)		sì	no	No
SecurityManagerPolitica (Nota 3)		sì	no	vuoto
SocksClient		sì	sì	No
HostSocksProxy	SocksClient	sì	sì	vuoto
SocksProxySocksProxy	SocksClient	sì	sì	1080
SocksServer		sì	sì	No
Client SSL		sì	sì	No
AnelloSSLClientCAKey	Client SSL	sì	sì	vuoto
SSLClientCAKeyRingPW	Client SSL	sì	sì	vuoto
“SSLClientCAKeyRingUseCryptoHardware” a pagina 254	Client SSL	sì	sì	No
SuiteSSLClientCipher	Client SSL	sì	sì	vuoto
Timeout diSSLClientConnect	Client SSL	sì	sì	30
SSLClientCustomOutboundSNI	Client SSL	sì	sì	vuoto
DN_C client SSL	Client SSL	sì	sì	* (Nota 1)
SSLClientDN_CN	Client SSL	sì	sì	* (Nota 1)
DN_DC client SSL	Client SSL	sì	sì	* (Nota 1)
DNQ client SSL	Client SSL	sì	sì	* (Nota 1)
DN client SSL	Client SSL	sì	sì	* (Nota 1)
DN_SSLClient_O	Client SSL	sì	sì	* (Nota 1)
OU client SSL	Client SSL	sì	sì	* (Nota 1)
PC client SSL	Client SSL	sì	sì	* (Nota 1)
SSLClientDN_ST	Client SSL	sì	sì	* (Nota 1)
SSLClientDN_Via	Client SSL	sì	sì	* (Nota 1)
DN client SSL	Client SSL	sì	sì	* (Nota 1)
UID_DNClientSSL	Client SSL	sì	sì	* (Nota 1)

Nome della proprietà	Proprietà da impostare true	Globale	Instradamento	Valore predefinito
SSLClientExit		sì	sì	No
AnelloSSLClientKey	Client SSL	sì	sì	vuoto
SSLClientKeyRingPW	Client SSL	sì	sì	vuoto
“SSLClientKeyRingUseCryptoHardware” a pagina 257	Client SSL	sì	sì	No
“SNI SSLClientOutbound” a pagina 257	Client SSL	sì	sì	Nome Host
SSLClientProtocols	Client SSL	sì	sì	TLSv1.2 TLSv1.3
SSLClientSiteDN_C	Client SSL	sì	sì	* (Nota 1)
SSLClientSiteDN_CN	Client SSL	sì	sì	* (Nota 1)
SSLClientSiteDN_DC	Client SSL	sì	sì	* (Nota 1)
SSLClientSiteDN_DNQ	Client SSL	sì	sì	* (Nota 1)
SSLClientSiteDN_L	Client SSL	sì	sì	* (Nota 1)
SSLClientSiteDN_O	Client SSL	sì	sì	* (Nota 1)
SSLClientSiteOU_DN	Client SSL	sì	sì	* (Nota 1)
SSLClientSiteDN_PC	Client SSL	sì	sì	* (Nota 1)
SSLClientSiteDN_ST	Client SSL	sì	sì	* (Nota 1)
SSLClientSiteDN_Street	Client SSL	sì	sì	* (Nota 1)
SSLClientSiteDN_T	Client SSL	sì	sì	* (Nota 1)
SSLClientSiteUID_DN	Client SSL	sì	sì	* (Nota 1)
EtichettaSSLClientSite	Client SSL	sì	sì	vuoto
SSLCommandPort		sì	no	vuoto
SSLCommandPortCipherSuites		sì	no	vuoto
SSLCommandPortListenerAddress		sì	no	vuoto
SSLCommandPortKeyRing		sì	no	vuoto
SSLCommandPortKeyRingPW		sì	no	vuoto
SSLCommandPortKeyRingUseCryptoHardware		sì	no	No
ProtocolliSSLCommandPort		sì	no	TLSv1.2 TLSv1.3
SSLCommandPortSiteLabel		sì	no	vuoto
SSLExitData	SSLServerExit	sì	sì	vuoto
SSLExitName	SSLServerExit	sì	sì	vuoto
SSLExitPath	SSLServerExit	sì	sì	<i>mqipt_home</i> <i>\ exits</i>
SSLExitTimeout	SSLServerExit	sì	sì	30

Nome della proprietà	Proprietà da impostare true	Globale	Instradamento	Valore predefinito
SSLProxyMode		sì	sì	No
SSLPlainConnections	SSLServer o SSLProxyMode	sì	sì	No
Server SSL		sì	sì	No
SSLServerAskClientAuth	Server SSL	sì	sì	No
AnelloSSLServerCAKey	Server SSL	sì	sì	vuoto
SSLServerCAKeyRingPW	Server SSL	sì	sì	vuoto
“SSLServerCAKeyRingUseCryptoHardware” a pagina 261	Server SSL	sì	sì	No
SuiteSSLServerCipher	Server SSL	sì	sì	vuoto
SSLServerDN_C	Server SSL	sì	sì	* (Nota 1)
SSLServerDN_CN	Server SSL	sì	sì	* (Nota 1)
DN_DC server SSL	Server SSL	sì	sì	* (Nota 1)
DNQ server SSL	Server SSL	sì	sì	* (Nota 1)
SSLServerDN_L	Server SSL	sì	sì	* (Nota 1)
SSLServerDN_O	Server SSL	sì	sì	* (Nota 1)
OU server SSL	Server SSL	sì	sì	* (Nota 1)
SSLServerDN_PC	Server SSL	sì	sì	* (Nota 1)
SSLServerDN_ST	Server SSL	sì	sì	* (Nota 1)
SSLServerDN_Via	Server SSL	sì	sì	* (Nota 1)
SSLServerDN_T	Server SSL	sì	sì	* (Nota 1)
UID server SSL	Server SSL	sì	sì	* (Nota 1)
SSLServerExit		sì	sì	No
SSLServerKeyAnello	Server SSL	sì	sì	vuoto
SSLServerKeyRingPW	Server SSL	sì	sì	vuoto
“SSLServerKeyRingUseCryptoHardware” a pagina 264	Server SSL	sì	sì	No
SSLServerProtocols	Server SSL	sì	sì	TLSv1.2 TLSv1.3
SSLServerSiteDN_C	Server SSL	sì	sì	* (Nota 1)
SSLServerSiteDN_CN	Server SSL	sì	sì	* (Nota 1)
SSLServerSiteDN_DC	Server SSL	sì	sì	* (Nota 1)
SSLServerSiteDN_DNQ	Server SSL	sì	sì	* (Nota 1)
SSLServerSiteDN_L	Server SSL	sì	sì	* (Nota 1)
SSLServerSiteDN_O	Server SSL	sì	sì	* (Nota 1)
SSLServerSiteUO_DN	Server SSL	sì	sì	* (Nota 1)

Nome della proprietà	Proprietà da impostare true	Globale	Instradamento	Valore predefinito
SSLServerSiteDN_PC	Server SSL	sì	sì	* (Nota 1)
SSLServerSiteDN_ST	Server SSL	sì	sì	* (Nota 1)
SSLServerSiteDN_Street	Server SSL	sì	sì	* (Nota 1)
SSLServerSiteDN_T	Server SSL	sì	sì	* (Nota 1)
SSLServerSiteUID_DN	Server SSL	sì	sì	* (Nota 1)
SSLServerSiteSSLServerSite	Server SSL	sì	sì	vuoto
StoredCredentialsStoredCredentials		sì	sì	vuoto
TCPKeepAlive		sì	sì	No
traccia		sì	sì	0
 TraceFilefile di traccia		sì	no	25
 TraceFile		sì	no	200
“Dati TraceUser” a pagina 267		sì	sì	64
UriName	HTTP	sì	sì	(Nota 2)

Note:

1. L'asterisco (*) rappresenta un carattere jolly.
2. Per i dettagli sulle impostazioni predefinite, consultare [UriName](#) in “MQIPT Proprietà instradamento” a pagina 247 .
3.  Questa proprietà è obsoleta per la rimozione in una release futura.

Riferimenti correlati

“IBM MQ Internet Pass-Thru Riferimento di configurazione” a pagina 237

IBM MQ Internet Pass-Thru (MQIPT) usa un file di configurazione denominato `mqipt.conf` per definire gli instradamenti e controllare le azioni del server MQIPT . Da IBM MQ 9.2, è possibile specificare anche le proprietà di configurazione per il comando `mqiptAdmin` in un file delle proprietà.

“MQIPT Proprietà globali” a pagina 243

Il file di configurazione `mqipt.conf` può contenere una serie di proprietà globali.

“MQIPT Proprietà instradamento” a pagina 247

Il file di configurazione `mqipt.conf` può contenere proprietà per singoli instradamenti.

MQIPT Proprietà globali

Il file di configurazione `mqipt.conf` può contenere una serie di proprietà globali.

Le proprietà riportate di seguito possono essere visualizzate solo nella sezione `[global]` di `mqipt.conf`. Tutte le [proprietà di instradamento](#) tranne **ListenerPort**, **Destination**, **DestinationPort**, **Name** e **OutgoingPort** possono essere visualizzate nella sezione `[global]` . Se una proprietà viene visualizzata sia nelle sezioni di instradamento che `[global]` , il valore della proprietà nella sezione `[route]` sovrascrive il valore globale, ma solo per l'instradamento in questione. In tal modo, la sezione `[global]` può essere utilizzata per stabilire i valori predefiniti da utilizzare per quelle proprietà che non sono impostate nelle singole sezioni `[route]` .

AccessPW

La parola d'ordine utilizzata per autenticare i comandi inviati alla porta comandi MQIPT utilizzando il comando `mqiptAdmin` .

Il valore può essere una password che è stata codificata utilizzando il comando **mqiPTPW** o una password in testo semplice. Le password in testo semplice possono contenere solo caratteri alfanumerici. Si consiglia di codificare le password memorizzate nella configurazione di MQIPT . Per ulteriori informazioni sulla codifica delle password nella configurazione di MQIPT , consultare [Codifica delle password memorizzate](#).

L'autenticazione viene eseguita per i comandi di gestione ricevuti dalla porta comandi se si verificano entrambe le condizioni riportate di seguito:

- La proprietà **AccessPW** viene specificata e impostata su un valore non vuoto.
- La proprietà **RemoteCommandAuthentication** viene specificata e impostata su un valore diverso da none.

CommandPort

Il numero di porta TCP/IP della porta comandi non protetta. MQIPT accetta i comandi di gestione inviati dal comando **mqiPTAdmin** a questa porta comandi.

Le connessioni alla porta comandi non protetta non vengono protette con TLS. I dati inviati alla porta comandi, inclusa la password di accesso, potrebbero essere acceduti da altri utenti della rete. Per configurare una porta comandi protetta con TLS, impostare invece la proprietà **SSLCommandPort** .

Se la proprietà **CommandPort** non è specificata, MQIPT non ascolta i comandi di gestione sulla porta comandi non protetta. Per utilizzare il numero di porta predefinito, 1881, utilizzato per impostazione predefinita dal comando **mqiPTAdmin** , impostare **CommandPort** su 1881.

CommandPortListenerAddress

L'indirizzo del listener locale che deve essere utilizzato dalla porta comandi non protetta. Impostando l'indirizzo del listener locale è possibile limitare le connessioni in entrata alla porta comandi non protetta a quelle provenienti da una particolare interfaccia di rete. Il valore predefinito è ascoltare su tutte le interfacce di rete.

ConnectionLog

`true` o `false`. Quando `true`, MQIPT registra tutti i tentativi di connessione (riusciti o meno) nella sottodirectory `logs` e gli eventi di disconnessione nel file `mqiPTYYYYMMDDHHmmSS.log` (dove `YYYYMMDDHHmmSS` sono caratteri che rappresentano la data e l'ora correnti). Il valore predefinito di **ConnectionLog** è `true`. Quando questa proprietà viene modificata da `true` a `false`, MQIPT chiude il log di connessione esistente e ne crea uno nuovo. Il nuovo log viene utilizzato quando la proprietà viene reimpostata su `true`.

Funzioni EnableAdvanced

Impostare questa proprietà su `true` per confermare che le funzionalità avanzate che richiedono la titolarità IBM MQ Advanced, IBM MQ Advanced for z/OS, IBM MQ Advanced for z/OS VUE, o IBM MQ Appliance possono essere utilizzate da MQIPT. Se si dispone della titolarità appropriata, è possibile utilizzare le funzionalità avanzate in MQIPT. Se le funzionalità avanzate sono abilitate su un instradamento, il gestore code locale connesso utilizzando l'instradamento MQIPT deve avere anche la titolarità IBM MQ Advanced, IBM MQ Advanced for z/OS, IBM MQ Advanced for z/OS VUE, o IBM MQ Appliance . Gli instradamenti che utilizzano funzionalità avanzate non possono iniziare a meno che questa proprietà non sia impostata su `true`. Quando questa proprietà viene modificata da `true` a `false`, gli instradamenti che utilizzano le funzionalità avanzate vengono arrestati.

LocalAdmin

Specifica se è consentita la gestione locale senza una porta comandi. I comandi di gestione inviati dal comando **mqiPTAdmin** utilizzando l'amministrazione locale invece della porta comandi, non vengono accettati se questa proprietà è impostata su `false`.

I valori validi per questa proprietà sono `true` e `false`. Il valore predefinito è `True`.

MaxLogFileSize

La dimensione massima (specificata in KB) del file di log di connessione. Quando la dimensione del file aumenta al di sopra di questo massimo, viene eseguita una copia di backup (`mqiPT001.log`) e viene avviato un nuovo file. Vengono conservati solo due file di backup (`mqiPT001.log` e `mqiPT002.log`); ogni volta che il file di log principale si riempie, tutti i backup precedenti vengono cancellati. Il valore predefinito di **MaxLogFileSize** è 50; il valore minimo consentito è 5.

Autenticazione RemoteCommand

Specifica se i comandi di gestione ricevuti dalla porta comandi non protetta o dalla porta comandi TLS devono essere autenticati. I comandi vengono autenticati controllando che la password fornita corrisponda alla password specificata nella proprietà AccessPW . Il valore può essere uno dei seguenti:

Nessuna

Non viene eseguita alcuna autenticazione sui comandi immessi in una delle porte di comando. Gli utenti del comando **mciptAdmin** non devono immettere una password. Questo è il valore predefinito.

facoltativo

Gli utenti del comando **mciptAdmin** non devono fornire una password. Tuttavia, se viene fornita una password, deve essere valida.

obbligatorio

Gli utenti del comando **mciptAdmin** devono fornire una password valida con ogni comando immesso sulle porte del comando.

La proprietà ... **AccessPW** deve essere specificata anche per abilitare l'autenticazione per le porte di comando.

RemoteShut

Specifica se MQIPT può essere arrestato da un comando di arresto inviato alla porta comandi non protetta o dalla porta comandi TLS dal comando **mciptAdmin** . Questa proprietà deve essere impostata su `true` per i comandi di arresto ricevuti da una delle porte comandi da elaborare.

I valori validi per questa proprietà sono `true` e `false`. Il valore predefinito è `False`.

SecurityManager

Impostare questa proprietà su `true` per abilitare Java security manager per questa istanza di MQIPT. È necessario assicurarsi che siano concesse le autorizzazioni corrette. Per ulteriori informazioni, consultare [Java security manager](#) . Il valore predefinito per questa proprietà è `false`.

Questa proprietà è obsoleta per la rimozione in una release futura.

Politica SecurityManager

Il nome file completo di un file delle politiche Java security manager . Se questa proprietà non è impostata, vengono utilizzati solo i file di politica utente e di sistema predefiniti. Se il Java security manager è già abilitato, le modifiche a questa proprietà non avranno effetto fino a quando Java security manager non sarà stato disabilitato e riabilitato.

 Questa proprietà è obsoleta per la rimozione in una release futura.

SSLCommandPort

Il numero di porta TCP/IP della porta comandi TLS. MQIPT accetta i comandi di gestione inviati dal comando **mciptAdmin** a questa porta comandi. Questa porta accetta solo connessioni TLS. Questa proprietà deve essere specificata per abilitare la porta comandi TLS.

SSLCommandPortCipherSuites

Il nome delle suite di cifratura da abilitare sulla porta comandi TLS. È possibile specificare più di una suite di cifratura separando i valori con virgole. È possibile specificare solo suite di cifratura TLS 1.2 e TLS 1.3 abilitate per impostazione predefinita nel Java runtime environment (JRE) fornito con MQIPT . Se questa proprietà non viene specificata, tutte le suite di cifratura abilitate in JRE vengono abilitate sulla porta comandi TLS.

SSLCommandPortListenerAddress

L'indirizzo del listener locale che deve essere utilizzato dalla porta comandi TLS. Impostando l'indirizzo del listener locale, è possibile limitare le connessioni in entrata alla porta comandi TLS a quelle da una particolare interfaccia di rete. Il valore predefinito è ascoltare su tutte le interfacce di rete.

SSLCommandPortKeyRing

Il nome del file keyring PKCS#12 che contiene il certificato server della porta comandi TLS.

Su piattaforme Windows , è necessario utilizzare una doppia barra rovesciata (\\) come separatore di file.

PW SSLCommandPortKeyRing

La password codificata per accedere al file keyring della porta comandi TLS o al keystore PKCS #11 . La password deve essere codificata utilizzando il comando mqiptPW e il valore di questa proprietà deve essere impostato sull'output della stringa da mqiptPW.

SSLCommandPortKeyRingUseCryptoHardware

Specifica se l'hardware crittografico che supporta l'interfaccia PKCS #11 viene utilizzato come keystore per il certificato del server della porta comandi TLS. I valori validi per questa proprietà sono true e false. Se questa proprietà è impostata su true, non è possibile specificare anche **SSLCommandPortKeyRing** .

L'utilizzo dell'hardware crittografico in MQIPT è una funzione IBM MQ Advanced . La proprietà EnableAdvancedCapabilities deve essere impostata su true per confermare che si dispone della titolarità IBM MQ Advanced .

Protocolli SSLCommandPort

Un elenco separato da virgole di protocolli da abilitare sulla porta comandi TLS. È possibile specificare uno o più dei seguenti valori.

Tabella 117. Valori consentiti per i protocolli TLS della porta comandi	
Valore	Protocollo
TLSv1.2	TLS 1.2
TLSv1.3	TLS 1.3

Se non si specifica questa proprietà, TLS 1.2 e TLS 1.3 sono abilitati per impostazione predefinita.

SSLCommandPortSiteLabel

Il nome etichetta del certificato server utilizzato dalla porta comandi TLS. Se questa proprietà non viene specificata, viene selezionato qualsiasi certificato nel keystore della porta comandi TLS compatibile con la suite di cifratura.

Traccia

Il livello di traccia per i thread MQIPT globali non associati a un instradamento e per gli instradamenti che non hanno una proprietà **Trace** impostata. Ad esempio, il thread di controllo MQIPT principale e i thread del server dei comandi non sono associati a un instradamento e vengono tracciati solo se la traccia è abilitata nella sezione [global] . Il valore della proprietà **Trace** in una sezione [route] sovrascrive la proprietà globale **Trace** per tale instradamento. Per informazioni sulla traccia dei thread associati a un instradamento, consultare la sezione **Trace** nella sezione [route] .

Il valore di questa proprietà può essere uno dei seguenti:

0

La traccia non è abilitata

Qualsiasi numero intero positivo

La traccia è abilitata

Il valore predefinito è 0.

V 9.4.0 Conteggio TraceFile

Il numero di file di traccia nella serie di file di rotazione utilizzati da MQIPT per scrivere i dati di traccia.

Il valore minimo consentito è 3. Il valore predefinito è 25.

Se si modifica il valore di questa proprietà, il file di traccia corrente viene chiuso e viene aperto il successivo file nella serie di rotazione dei file di traccia.

V 9.4.0 TraceFile

La dimensione massima dei file di traccia prodotti da MQIPT, specificata in MB.

Il valore minimo consentito è 1. Il valore predefinito è 200.

Se si modifica il valore di questa proprietà, il file di traccia corrente viene chiuso e viene aperto il successivo file nella serie di rotazione dei file di traccia.

MQIPT Proprietà instradamento

Il file di configurazione `mqipt.conf` può contenere proprietà per singoli instradamenti.

La `[route]` sezione del file di configurazione `mqipt.conf` può contenere le seguenti proprietà:

Attivo

L'instradamento accetta le connessioni in entrata solo se il valore di **Active** è impostato su `true`. Ciò significa che è possibile chiudere temporaneamente l'accesso alla destinazione, impostando questo valore su `false`, senza dover eliminare la sezione `[route]` dal file di configurazione. Se si modifica questa proprietà in `false`, l'instradamento viene arrestato quando viene emesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

V9.4.0 V9.4.0 AllowedProtocols

Specifica i protocolli accettati da questo instradamento. Le connessioni che utilizzano un protocollo non specificato in questo elenco vengono rifiutate. È possibile specificare più di un protocollo come elenco separato da virgole. È possibile specificare i protocolli seguenti:

mq

L'instradamento accetta le connessioni che utilizzano il protocollo IBM MQ.

http

La rotta accetta le connessioni HTTP da un'altra istanza di MQIPT.

Il valore predefinito di questa proprietà è `mq`.

Se il valore di questa proprietà viene modificato, l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

ClientAccess

L'instradamento consente le connessioni del canale client in entrata solo se il valore di **ClientAccess** è impostato su `true`. Tenere presente che, potenzialmente, è possibile configurare MQIPT in modo che accetti solo le richieste client, solo le richieste del gestore code o entrambi i tipi di richiesta. Utilizzare questa proprietà insieme alla proprietà **QMgrAccess**. Se si modifica questa proprietà in `false`, l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

Destinazione

Il nome host (o l'indirizzo IP decimale puntato) del gestore code o dell'istanza MQIPT successiva a cui deve connettersi questo instradamento. Ogni sezione `[route]` deve contenere un valore **Destination** esplicito, ma diverse sezioni `[route]` possono fare riferimento alla stessa destinazione. Se una modifica a questa proprietà influisce su un instradamento, l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate. Quando si utilizza la proprietà **SocksProxyHost**, la proprietà **Destination** deve utilizzare il formato di indirizzo IPv4 decimale puntato.

DestinationPort

La porta sull'host di destinazione a cui deve connettersi questo instradamento. Ogni sezione `[route]` deve contenere un valore **DestinationPort** esplicito, ma diversi percorsi possono fare riferimento alla stessa combinazione di valori **Destination** e **DestinationPort**. Se una modifica a questa proprietà influisce su un instradamento, l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

HTTP

Impostare **HTTP** su `true` per gli instradamenti responsabili delle richieste di tunneling HTTP in uscita. La proprietà **Destination** per la rotta deve essere il nome host di un'altra istanza di MQIPT quando **HTTP** è impostata su `true`. Impostare **HTTP** su `false` per gli instradamenti connessi ai gestori code

IBM MQ . Se si modifica questa proprietà, l'instradamento viene arrestato. Almeno una delle proprietà **HTTPProxy** o **HTTPServer** deve essere specificata anche quando **HTTP** è impostato su `true`. Questa proprietà non può essere utilizzata insieme alla proprietà **SocksClient** .

V 9.4.0 V 9.4.0 HTTPConnectionTimeout

Questa proprietà specifica il tempo, espresso in millesimi di secondo, per cui MQIPT attende che una connessione HTTP ricevuta venga stabilita correttamente, prima che la connessione venga rifiutata.

Il valore predefinito è 5000.

Proxy HTTP

Il nome host (o indirizzo IP decimale puntato) del proxy HTTP utilizzato da tutte le connessioni per questo instradamento. Una richiesta **CONNECT** viene inviata al proxy HTTP, invece della richiesta **POST** che viene normalmente utilizzata quando non viene configurato alcun proxy HTTP. Se si modifica questa proprietà (e **HTTP** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

HTTPProxyPort

L'indirizzo della porta da utilizzare sul proxy HTTP. Il valore predefinito è 8080. Se si modifica questa proprietà (e **HTTP** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

Server HTTP

Il nome host (o l'indirizzo IP decimale puntato) del server HTTP utilizzato da tutte le connessioni per questo instradamento. Generalmente è il nome host di un'altro MQIPT.

Se **HTTPProxy** non viene specificato, MQIPT si collega all'host specificato in **HTTPServer** ed emette richieste HTTP **POST** all'host specificato nella proprietà **Destination** di instradamento. Se viene specificato **HTTPProxy** , MQIPT si connette all'host specificato in **HTTPProxy** e richiede che il proxy stabilisca un tunnel per l'host specificato in **HTTPServer**.

Se viene specificato **HTTPProxy** , il valore predefinito è l'instradamento **Destination**.

Se si modifica questa proprietà (e **HTTP** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

HTTPS

Impostare **HTTPS** su `true` per effettuare richieste HTTPS. Anche le proprietà **HTTP** e **SSLClient** devono essere abilitate e il key ring del client deve essere configurato utilizzando la proprietà **SSLClientKeyRing** o **SSLClientKeyRingUseCryptoHardware** , come per l'operazione SSL/TLS. Se si modifica la proprietà **HTTPS** (e **HTTP** è impostato su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

HTTPServerPort

L'indirizzo della porta da utilizzare sul server HTTP. Il valore predefinito è 8080, a meno che non venga specificato **HTTPProxy** , nel qual caso il valore predefinito è l'instradamento **DestinationPort**.

Se si modifica questa proprietà (e **HTTP** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

IdleTimeout

Il tempo, in minuti, dopo il quale viene chiusa una connessione inattiva. Notare che i canali da gestore code a gestore code hanno anche la proprietà **DISCINT** . Se si imposta il parametro **IdleTimeout** , prendere nota di **DISCINT**. Se **IdleTimeout** è impostato su 0, non vi è alcun timeout di inattività. Le modifiche a questa proprietà diventano effettive solo quando l'instradamento viene riavviato.

CRL IgnoreExpired

Impostare **IgnoreExpiredCRLs** su `true` per ignorare un CRL scaduto. Il valore predefinito è `False`. Tieni presente che se imposti **IgnoreExpiredCRLs** su `true`, potrebbe essere utilizzato un certificato revocato per stabilire una connessione SSL/TLS.

LDAP

Impostare **LDAP** su `true` per abilitare l'utilizzo di un server LDAP quando si utilizzano connessioni SSL/TLS. MQIPT utilizzerà il server LDAP per richiamare CRL e ARL. Anche la proprietà **SSLClient** o **SSLServer** deve essere impostata su `true` per rendere effettiva questa proprietà.

LDAPCacheTimeout

Il tempo di scadenza, in ore, della cache temporanea in cui viene archiviato un CRL richiamato da un server LDAP. Trascorso questo tempo, l'intera cache CRL viene svuotata. Ad esempio, specificare un valore di 1 ora significa che la cache viene svuotata una volta all'ora. Il valore predefinito è 24. Se si specifica un valore di timeout di 0, le voci nella cache non scadranno fino al riavvio dell'instradamento. Se si modifica questa proprietà (e **LDAP** è impostato su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

LDAPIgnoreErrors

Impostare **LDAPIgnoreErrors** su `true` per ignorare eventuali errori di connessione o di timeout quando si esegue una ricerca LDAP. Se MQIPT non può eseguire una ricerca riuscita, non consentirà il completamento della connessione client, a meno che questa proprietà non sia stata abilitata. Una ricerca riuscita indica che un CRL è stato richiamato o che non ci sono CRL disponibili per la CA specificata. Se si modifica questa proprietà (e **LDAP** è impostato su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

Nota: Se si abilita questa proprietà, è possibile utilizzare un certificato revocato per stabilire una connessione SSL/TLS.

LDAPServer1

Il nome host o l'indirizzo IP del server LDAP principale. Questa proprietà deve essere impostata se LDAP è stato impostato su `true`. Se si modifica questa proprietà (e **LDAP** è impostato su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

LDAPServer1Port

Il numero di porta di ascolto del server principale LDAP. Il valore predefinito è 389. Se si modifica questa proprietà (e **LDAP** è impostato su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

LDAPServer1Userid

L'ID utente necessario per accedere al server LDAP principale. Questa proprietà deve essere impostata se è richiesta l'autorizzazione per accedere al server LDAP principale. Se si modifica questa proprietà (e **LDAP** è impostato su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

LDAPServer1Password

La password necessaria per accedere al server LDAP principale. Questa proprietà deve essere impostata se **LDAPServer1Userid** è stato impostato su `true`. Se si modifica questa proprietà (e **LDAP** è impostato su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

Il valore può essere una password che è stata codificata utilizzando il comando **mqiptPW** o una password in testo semplice. Le password in testo semplice possono contenere solo caratteri alfanumerici. Si consiglia di codificare le password memorizzate nella configurazione di MQIPT. Per ulteriori informazioni sulla codifica delle password nella configurazione di MQIPT, consultare [Codifica delle password memorizzate](#).

LDAPServer1Timeout

Il tempo, in secondi, che MQIPT attende per una risposta dal server LDAP principale. Il valore predefinito è 0, che indica che la connessione non andrà in timeout. Se si modifica questa proprietà (e **LDAP** è impostato su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

LDAPServer2

Il nome host o l'indirizzo IP del server LDAP di backup. Questa proprietà è facoltativa. Se si modifica questa proprietà (e **LDAP** è impostato su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

LDAPServer2Port

Il numero di porta di ascolto del server LDAP di backup. Il valore predefinito è 389. Se si modifica questa proprietà (e **LDAP** è impostato su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

LDAPServer2Userid

L'id utente necessario per accedere al server LDAP di backup. Questa proprietà deve essere impostata se è richiesta l'autorizzazione per accedere al server LDAP di backup. Se si modifica questa proprietà (e **LDAP** è impostato su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

LDAPServer2Password

La password necessaria per accedere al server LDAP di backup. Questa proprietà deve essere impostata se **LDAPServer2** è stato impostato su `true`. Se si modifica questa proprietà (e **LDAP** è impostato su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

Il valore può essere una password che è stata codificata utilizzando il comando **mqiptPW** o una password in testo semplice. Le password in testo semplice possono contenere solo caratteri alfanumerici. Si consiglia di codificare le password memorizzate nella configurazione di MQIPT. Per ulteriori informazioni sulla codifica delle password nella configurazione di MQIPT, consultare [Codifica delle password memorizzate](#).

LDAPServer2Timeout

Il tempo, in secondi, che MQIPT attenderà una risposta dal server LDAP di backup. Il valore predefinito è 0, che indica che la connessione non andrà in timeout. Se si modifica questa proprietà (e **LDAP** è impostato su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

ListenerAddress

Utilizzare questa proprietà se il sistema MQIPT ha più indirizzi IP ed è necessario collegare la porta del listener di instradamento ad un indirizzo specifico. Ciò è utile per limitare le connessioni in entrata a quelle da una particolare interfaccia di rete. Il valore di questa proprietà deve essere un indirizzo IP appartenente a una delle interfacce di rete sul sistema su cui è in esecuzione MQIPT. Il valore predefinito è di accettare connessioni da tutte le interfacce di rete.

ListenerPort

Il numero di porta su cui l'instradamento deve essere in ascolto delle richieste in entrata. Ogni sezione [route] deve contenere un valore **ListenerPort** esplicito. I valori **ListenerPort** impostati in ogni sezione devono essere distinti. È possibile utilizzare qualsiasi numero di porta valido, incluse le porte 80 e 443, purché le porte scelte non siano già utilizzate da altri listener TCP/IP in esecuzione sullo stesso host.

LocalAddress

L'indirizzo IP a cui collegare tutte le connessioni per questo instradamento su questo computer. L'indirizzo scelto deve essere un indirizzo IP associato a una delle interfacce di rete sul computer su cui è in esecuzione MQIPT. Se si modifica questa proprietà, l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

Thread MaxConnection

Il numero massimo di thread di connessione e quindi il numero massimo di connessioni simultanee che possono essere gestite da questo instradamento. Se questo limite viene raggiunto, il valore **MaxConnectionThreads** indica anche il numero di connessioni accodate quando tutti i thread sono in uso. Oltre tale numero, le successive richieste di connessione vengono rifiutate.

Il valore minimo consentito è il maggiore di 1 e il valore di **MinConnectionThreads**.

Se il valore viene aumentato, il nuovo valore viene utilizzato quando viene immesso il comando di aggiornamento. Tutte le connessioni utilizzano il nuovo valore immediatamente. L'instradamento non è arrestato.

Se il valore viene diminuito, il nuovo valore diventa effettivo solo quando l'instradamento viene riavviato.

Thread MinConnection

Il numero di thread di connessione assegnati per gestire le connessioni in entrata su un instradamento quando l'instradamento viene avviato. Il numero di thread assegnati non scende al di sotto di questo valore durante il periodo di tempo in cui l'instradamento è attivo.

Il valore deve essere compreso nell'intervallo tra 0 e **MaxConnectionThreads**.

Le modifiche a questa proprietà diventano effettive solo quando l'instradamento viene riavviato.

Nome

Un nome che consente di identificare l'instradamento. Questa proprietà è facoltativa. Il valore viene visualizzato nei messaggi della console e nelle informazioni di traccia. Le modifiche a questa proprietà diventano effettive solo quando l'instradamento viene riavviato.

OutgoingPort

Il numero di porta iniziale utilizzato dalle connessioni in uscita. L'intervallo di numeri porta corrisponde al valore **MaxConnectionThread** per questo instradamento. Il valore predefinito 0 utilizza un numero di porta definito dal sistema. Se si modifica questa proprietà, l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate. Quando viene utilizzato HTTP, ogni connessione canale richiede due porte in uscita. Per ulteriori informazioni, consultare [Controllo del numero di porta](#).

V 9.4.0

PasswordProtection

Specifica se MQIPT è in grado di aggiungere o rimuovere la protezione per le credenziali inviate nelle strutture MQCSP da IBM MQ clients, al fine di conservare la compatibilità tra il client e il gestore code, per le rotte MQIPT configurate per aggiungere o rimuovere la crittografia TLS.

Le credenziali nelle strutture MQCSP possono essere protette, utilizzando la funzione di protezione della password MQCSP IBM MQ o crittografate utilizzando la crittografia TLS. La protezione della password MQCSP è utile per scopi di test e sviluppo in quanto è più semplice rispetto all'impostazione della cifratura TLS, ma non è così sicura.

Per ulteriori informazioni sulla protezione della password MQCSP, consultare [Protezione della password MQCSP](#).

Quando una rotta MQIPT è configurata per l'aggiunta o la rimozione della codifica TLS, MQIPT potrebbe dover proteggere le credenziali nella struttura MQCSP o rimuovere la protezione della password MQCSP perché la connessione abbia esito positivo.

Il valore della proprietà può essere uno dei seguenti:

obbligatorio

MQIPT garantisce che le credenziali nella struttura MQCSP siano codificate utilizzando TLS o protette con la protezione della password MQCSP.

Se le credenziali nella struttura MQCSP vengono inviate codificate dal client utilizzando la crittografia TLS e la rotta MQIPT rimuove la crittografia TLS, MQIPT protegge le credenziali con la protezione della password MQCSP prima di inoltrare le credenziali alla destinazione della rotta. Ciò si verifica quando la rotta MQIPT è configurata con `SSLServer=true` e `SSLClient=false` e la CipherSuite selezionata non utilizza un codice null.

Se le credenziali nella struttura MQCSP sono protette dal client con la protezione password MQCSP, MQIPT non rimuove la protezione, anche se la connessione tra MQIPT e la destinazione della rotta utilizza la codifica TLS. Se la connessione tra MQIPT e la destinazione della rotta utilizza la crittografia TLS, la connessione potrebbe non riuscire con codice motivo MQRC_PASSWORD_PROTECTION_ERROR (2594).

Questo è il valore predefinito.

compatibile

MQIPT applica o rimuove la protezione della password MQCSP come richiesto per garantire che la connessione abbia esito positivo.

Se le credenziali nella struttura MQCSP vengono inviate crittografate dal client utilizzando la crittografia TLS e la rotta MQIPT rimuove la crittografia TLS, MQIPT protegge le credenziali con la protezione della password MQCSP prima di inoltrare la password alla destinazione della rotta. Ciò si verifica quando la rotta MQIPT è configurata con `SSLServer=true` e `SSLClient=false` e la CipherSuite selezionata non utilizza un codice null.

Se le credenziali nella struttura MQCSP sono protette dal client con la protezione della parola d'ordine MQCSP e l' MQIPT instradamento aggiunge la crittografia TLS, MQIPT rimuove la protezione della password MQCSP prima di inoltrare le credenziali alla destinazione dell'instradamento. Ciò si verifica quando la rotta MQIPT è configurata con `SSLServer=false` e `SSLClient=true` e la CipherSuite selezionata non utilizza un codice null.

Questa opzione offre la migliore compatibilità. Tuttavia, deve essere utilizzato solo per scopi di test e sviluppo su reti attendibili, poiché non garantisce che la password sia protetta sulla rete.

passthru

Le credenziali nella struttura MQCSP vengono inoltrate alla destinazione dell'instradamento da MQIPT senza aggiungere o rimuovere la protezione della parola d'ordine MQCSP. Se l'instradamento MQIPT è configurato per aggiungere o rimuovere la cifratura TLS, le connessioni client potrebbero non riuscire con codice motivo MQRC_PASSWORD_PROTECTION_ERROR (2594).

QMgrAccess

Impostare **QMgrAccess** su `true` per consentire le connessioni del canale del gestore code in entrata (ad esempio, i canali mittente). Se si modifica questa proprietà in `false`, l'instradamento viene arrestato quando viene emesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate.

RouteRestart

Impostare **RouteRestart** su `false` per impedire il riavvio dell'instradamento quando sono state modificate altre proprietà dell'instradamento ed è stato immesso un comando di aggiornamento. Il valore predefinito per questa proprietà è `true`.

SecurityExit

Impostare **SecurityExit** su `true` per abilitare un'uscita di sicurezza definita dall'utente. Il valore predefinito per questa proprietà è `false`.

Nome SecurityExit

Il nome classe dell'uscita di sicurezza definita dall'utente. Questa proprietà deve essere impostata se **SecurityExit** è stato impostato su `true`. Se si modifica questa proprietà (e **SecurityExit** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate.

Percorso SecurityExit

Il nome percorso completo contenente l'uscita di sicurezza definita dall'utente. Se questa proprietà non è stata impostata, per impostazione predefinita verrà utilizzata la sottodirectory `exit`. Questa proprietà può anche definire il nome di un file JAR (Java archive) contenente l'uscita di sicurezza definita dall'utente. Se si modifica questa proprietà (e **SecurityExit** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate.

Timeout SecurityExit

Il valore di timeout (in secondi) utilizzato da MQIPT per stabilire per quanto tempo attendere una risposta quando si convalida una richiesta di connessione. Il valore predefinito è 30. Se si modifica questa proprietà (e **SecurityExit** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SocksClient

Impostare **SocksClient** su `true` per fare in modo che la rotta agisca come un client SOCKS e definire tutte le connessioni tramite il proxy SOCKS con le proprietà **SocksProxyHost** e

SocksProxyPort . Se si modifica questa proprietà, l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate. Questa proprietà non può essere utilizzata con:

- **HTTP**
- **SocksServer**
- **SSLClient**
- **SSLProxyMode**

Host SocksProxy

Il nome host (o l'indirizzo IPv4 decimale puntato) del proxy SOCKS utilizzato da tutte le connessioni per questo instradamento. Se si modifica questa proprietà (e **SocksClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate. Quando si utilizza la proprietà **SocksProxyHost** , la proprietà **Destination** deve utilizzare il formato decimale puntato.

Porta SocksProxy

Il numero di porta da utilizzare su un proxy SOCKS. Il valore predefinito è 1080. Se si modifica questa proprietà (e **SocksClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SocksServer

Impostare **SocksServer** su `true` per fare in modo che l'instradamento agisca come un proxy SOCKS e accetti le connessioni client SOCKS. Se si modifica questa proprietà, l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate. Questa proprietà non può essere utilizzata con le seguenti proprietà:

- **SocksClient**
- **SSLProxyMode**
- **SSLServer**

Client SSL

Imposta **SSLClient** su `true` per fare in modo che la rotta agisca come un client SSL/TLS e stabiliscano connessioni SSL/TLS in uscita. L'impostazione di **SSLClient** su `true` implica che la destinazione sia un'altra istanza di MQIPT che funge da server SSL/TLS o da proxy / server HTTP.

Se si imposta **SSLClient** su `true`, è necessario specificare un file di chiavi del client SSL/TLS utilizzando la proprietà **SSLClientKeyRing** o **SSLClientCAKeyRing** oppure configurare MQIPT per utilizzare l'hardware crittografico impostando la proprietà **SSLClientKeyRingUseCryptoHardware** o **SSLClientCAKeyRingUseCryptoHardware** .

Se si modifica **SSLClient**, l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate.

Questa proprietà non può essere utilizzata insieme alla seguente proprietà:

- **SSLProxyMode**

Anello SSLClientCAKey

Il nome file completo del file key ring contenente certificati CA, utilizzato per autenticare certificati dal server SSL/TLS. Su piattaforme Windows , è necessario utilizzare una doppia barra rovesciata (\\) come separatore di file. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientCAKeyRingPW

La password per aprire il file di chiavi CA del client SSL/TLS specificato con la proprietà **SSLClientCAKeyRing** o per connettersi all'archivio chiavi hardware di crittografia se la proprietà **SSLClientCAKeyRingUseCryptoHardware** è impostata su `true`.

Il valore può essere una password che è stata codificata utilizzando il comando **mqiptPW** oppure il nome file completo del file che contiene una password codificata. Se si specifica un nome file su piattaforme Windows , è necessario utilizzare una doppia barra rovesciata (\\) come separatore di file. Si consiglia di migrare qualsiasi parola d'ordine key ring attualmente memorizzata in un file per utilizzare il metodo di protezione più recente e più sicuro, codificando nuovamente le parole d'ordine utilizzando il programma di utilità **mqiptPW** . Per ulteriori informazioni sulla codifica delle password nella configurazione di MQIPT , consultare [Codifica delle password memorizzate](#).

Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientCAKeyRingUseCryptoHardware

Specifica se l'hardware di crittografia che supporta l'interfaccia PKCS #11 viene utilizzato come keystore per i certificati CA utilizzati per autenticare i certificati del server dal server SSL/TLS, quando MQIPT agisce come client SSL/TLS. Se questa proprietà è impostata su `true`, **SSLClientCAKeyRing** non può essere impostata sullo stesso instradamento.

Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

L'utilizzo dell'hardware crittografico con MQIPT è una funzionalità IBM MQ Advanced . Per utilizzare questa funzionalità, il gestore code locale connesso mediante l'instradamento MQIPT deve avere anche la titolarità IBM MQ Advanced, IBM MQ Appliance, IBM MQ Advanced for z/OS o IBM MQ Advanced for z/OS VUE . L'instradamento non verrà avviato quando questa proprietà è impostata su `true` a meno che la proprietà globale **EnableAdvancedCapabilities** non sia impostata per confermare che è possibile utilizzare le capacità IBM MQ Advanced .

Suite SSLClientCipher

Il nome della CipherSuite SSL/TLS da utilizzare sul lato client SSL/TLS. Può essere una o più delle CipherSuites supportate. Se si lascia questa proprietà vuota, viene utilizzata qualsiasi CipherSuite per i protocolli abilitati compatibile con il certificato del client nel file di chiavi. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate.

Timeout SSLClientConnect

Il tempo (in secondi) di attesa di una connessione SSL/TLS da parte di un client SSL/TLS.

Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientCustomOutboundSNI

Specifica il valore di SNI ([Server Name Indication](#)) quando MQIPT avvia una connessione TLS alla destinazione dell'instradamento, se l'instradamento è configurato con **SSLClientOutboundSNI** impostato su `personalizzato`. Utilizzare questa proprietà ... per impostare SNI su un valore specifico che non può essere impostato automaticamente da MQIPT. Ad esempio, se si desidera impostare lo SNI su un nome host, ma la destinazione dell'instradamento è configurata con un indirizzo IP.

Il valore deve essere un IDN (Internationalized Domain Name) valido conforme alla specifica RFC 3490 e non può terminare con un punto finale. L'instradamento non inizia se viene specificato un valore non valido.

Se si modifica il valore di questa proprietà e **SSLClientOutboundSNI** è impostato su `personalizzato`, l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento.



Attenzione: Non utilizzare questa impostazione quando si inoltrano le connessioni a un canale IBM MQ che ha un'etichetta di certificato configurata nel campo **CERTLABL** del canale. Se si inoltra un client in questo modo, questo verrà rifiutato con un codice di ritorno

MQRC_SSL_INITIALIZATION_ERROR e un errore AMQ9673 stampato nei log degli errori del gestore code remoto.

CDN client SSL

Utilizzare questa proprietà per accettare i certificati ricevuti dal server SSL/TLS che corrispondono a questo nome paese. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, vengono accettati tutti i nomi dei paesi. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

CN client SSL

Utilizzare questa proprietà per accettare i certificati ricevuti dal server SSL/TLS che corrispondono a questo nome comune. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, vengono accettati tutti i nomi comuni. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

DN_DC client SSL

Utilizzare questa proprietà per accettare i certificati ricevuti dal server SSL/TLS che corrispondono a questo componente del dominio. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. È possibile specificare più DC separandoli con virgole. Ogni DC rappresenta un elemento in un nome dominio, ad esempio il nome dominio `example.ibm.com` è rappresentato come `example,ibm,com` utilizzando virgole per separare i valori multipli. Se non si specifica questa proprietà, vengono accettati tutti i componenti del dominio. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

DNQ client SSL

Utilizzare questa proprietà per accettare i certificati ricevuti dal server SSL/TLS che corrispondono a questo qualificatore di dominio. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, vengono accettati tutti i qualificatori di dominio. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientDN_L

Utilizzare questa proprietà per accettare i certificati ricevuti dal server SSL/TLS che corrispondono a questa posizione. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, si implicano "tutte le ubicazioni". Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

DN client SSL O

Utilizzare questa proprietà per accettare i certificati ricevuti dal server SSL/TLS che corrispondono all'organizzazione. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati da tutte le organizzazioni. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

OU client SSL

Utilizzare questa proprietà per accettare i certificati ricevuti dal server SSL/TLS che corrispondono a questa OU (Organizational Unit). Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. È possibile specificare più OU separandole con virgole. (Corrisponde a una virgola letterale antepoendo un carattere barra retroversa (\).) La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono

accettati con qualsiasi nome OU. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate.

PC client SSL

Utilizzare questa proprietà per accettare i certificati ricevuti dal server SSL/TLS che corrispondono a questo codice postale. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, vengono accettati tutti i codici postali. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLDN_ST

Utilizzare questa proprietà per accettare i certificati ricevuti dal server SSL/TLS che corrispondono a questo stato. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati dai server in tutti gli stati. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientDN_Via

Utilizzare questa proprietà per accettare i certificati ricevuti dal server SSL/TLS che corrispondono a questo nome strada. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, vengono accettati tutti i nomi delle strade. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

DN client SSL T

Utilizzare questa proprietà per accettare i certificati ricevuti dal server SSL/TLS che corrispondono a questo titolo. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, vengono accettati tutti i titoli. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

UID_DN_SSLClient

Utilizzare questa proprietà per accettare i certificati ricevuti dal server SSL/TLS che corrispondono a questo ID utente. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, vengono accettati tutti gli ID utente. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientExit

Utilizzare questa proprietà per abilitare o disabilitare l'utilizzo di un'uscita quando l'instradamento funge da client SSL/TLS. Ciò consente di definire i dettagli di uscita nel file di configurazione senza che vengano effettivamente utilizzati.

Anello SSLClientKey

Il nome file completo del file key ring contenente il certificato client. Su piattaforme Windows , è necessario utilizzare una doppia barra rovesciata (\\) come separatore di file. Se si modifica **SSLClientKeyRing** (e **SSLClient** è impostata su `true`), la rotta viene arrestata e riavviata quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientKeyRingPW

La password per aprire il file keyring del client SSL/TLS specificato con la proprietà **SSLClientKeyRing** o per connettersi al keystore dell'hardware crittografico se la proprietà **SSLClientKeyRingUseCryptoHardware** è impostata su `true`.

Il valore può essere una password che è stata codificata utilizzando il comando **mqiPTW** oppure il nome file completo del file che contiene una password codificata. Se si specifica un nome file su piattaforme Windows , è necessario utilizzare una doppia barra rovesciata (\\) come separatore di

file. Si consiglia di migrare qualsiasi parola d'ordine key ring attualmente memorizzata in un file per utilizzare il metodo di protezione più recente e più sicuro, codificando nuovamente le parole d'ordine utilizzando il programma di utilità **mqiptPW**. Per ulteriori informazioni sulla codifica delle password nella configurazione di MQIPT, consultare [Codifica delle password memorizzate](#).

Se si modifica **SSLClientKeyRingPW** (e **SSLClient** è impostata su `true`), la rotta viene arrestata e riavviata quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientKeyRingUseCryptoHardware

Specifica se l'hardware crittografico che supporta l'interfaccia PKCS #11 viene utilizzato come keystore contenente il certificato client, quando MQIPT agisce come un client SSL/TLS. Se questa proprietà è impostata su `true`, **SSLClientKeyRing** non può essere impostata sullo stesso instradamento.

Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

L'utilizzo dell'hardware crittografico con MQIPT è una funzionalità IBM MQ Advanced. Per utilizzare questa funzionalità, il gestore code locale connesso mediante l'instradamento MQIPT deve avere anche la titolarità IBM MQ Advanced, IBM MQ Appliance, IBM MQ Advanced for z/OS o IBM MQ Advanced for z/OS VUE. L'instradamento non verrà avviato quando questa proprietà è impostata su `true` a meno che la proprietà globale **EnableAdvancedCapabilities** non sia impostata per confermare che è possibile utilizzare le capacità IBM MQ Advanced.

SNI SSLClientOutbound

Specifica il valore dell'estensione SNI ([Server Name Indication](#)) quando MQIPT avvia una connessione TLS alla destinazione dell'instradamento. Lo SNI viene utilizzato dai gestori code IBM MQ per presentare il certificato corretto durante l'handshake TLS o per instradare le connessioni alla destinazione, in base alla configurazione.

Questa proprietà è applicabile solo agli instradamenti definiti con `SSLClient=true` e non può essere specificata per gli instradamenti definiti con `HTTP=true`. Se si modifica il valore di questa proprietà e **SSLClient** è impostata su `true`, l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento.



Attenzione: Se il canale di destinazione è configurato con un'etichetta di certificato sul campo **CERTLABL** dell'oggetto canale, è necessario impostare l'impostazione **CERTLABL** sul valore del canale. Se un client viene inoltrato senza l'impostazione SNI del canale, viene rifiutato con un codice di ritorno `MQRC_SSL_INITIALIZATION_ERROR` e un messaggio `AMQ9673` stampato nei log degli errori del gestore code remoto.

Il valore della proprietà può essere uno dei seguenti:

nome host

SNI è impostato sul nome host della destinazione di instradamento. Utilizza questa opzione se l'instradamento si connette a un programma di bilanciamento del carico o a un router che utilizza SNI per instradare le richieste. Ad esempio, Red Hat® OpenShift® Container Platform Router utilizza SNI per instradare le richieste al gestore code IBM MQ.

Se la destinazione di instradamento è un gestore code, le richieste di connessione ricevono il certificato predefinito del gestore code remoto durante l'handshake TLS e quindi non è possibile utilizzare i certificati per canale.

Se la destinazione dell'instradamento viene specificata utilizzando un indirizzo IP e non è possibile eseguire una ricerca DNS inversa, lo SNI è vuoto.

Questo è il valore predefinito.

canale

SNI è impostata sul nome del canale IBM MQ. Utilizzare questa opzione per consentire l'uso dei certificati per canale da parte del gestore code di destinazione, se le connessioni ricevute dall'instradamento non contengono il nome canale nell'SNI per uno dei seguenti motivi:

- La rotta è configurata per accettare connessioni che non sono protette con TLS con `SSLServer=false` o `SSLPlainConnections=true`.
- L'applicazione che si connette all'instradamento non può impostare SNI o è configurata per impostare SNI su un valore diverso dal nome del canale IBM MQ .

passthru

Se l'instradamento è definito con `SSLServer=true`, l'SNI sulla connessione in entrata è impostato sul valore dell'SNI ricevuto sulla connessione in entrata all'instradamento. Se l'instradamento non è configurato per accettare le connessioni TLS, SNI è impostato sul nome host di destinazione.

personalizzato

SNI è impostata sul valore specificato nella proprietà **SSLClientCustomOutboundSNI** . Se la proprietà **SSLClientCustomOutboundSNI** non viene specificata, SNI viene impostato come se l'instradamento fosse configurato con `SSLClientOutboundSNI=hostname`.

Nessuna

SNI non è impostato.

SSLClientProtocols

Utilizzato per limitare la serie di protocolli socket sicuri abilitati utilizzati per effettuare connessioni in uscita alla destinazione per un instradamento quando **SSLClient** è impostato su `true`.

È possibile specificare più valori separandoli con virgole. Se non si specifica questa proprietà, TLS 1.2 e TLS 1.3 sono abilitati per impostazione predefinita. Per abilitare protocolli diversi da TLS 1.2 o TLS 1.3, devi specificare i protocolli da abilitare in questa proprietà e aggiungere anche il supporto per il protocollo in Java runtime environment seguendo la procedura in [Enabling deprecated protocols and CipherSuites](#). È possibile specificare uno o più dei seguenti valori.

<i>Tabella 118. Valori consentiti per protocolli SSL/TLS</i>	
Valore	Protocollo
SSLv3	SSL 3.0
TLSv1	TLS 1.0
TLSv1.1	TLS 1.1
TLSv1.2	TLS 1.2
TLSv1.3	TLS 1.3

Utilizzare la voce elencata nella colonna **Valore** nella proprietà `route`. La voce corrispondente nella colonna **Protocollo** è solo a scopo informativo.

SSLClientSiteDN_C

Utilizzare questa proprietà per specificare un nome paese per selezionare un certificato da inviare al server SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome paese. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientSiteDN_CN

Utilizzare questa proprietà ... per specificare un nome comune per selezionare un certificato da inviare al server SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome comune. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientSiteDN_DC

Utilizzare questa proprietà per specificare un nome componente di dominio per selezionare un certificato da inviare al server SSL/TLS. La corrispondenza del certificato non è sensibile al

maiuscolo / minuscolo. È possibile specificare più DC separandoli con virgole. Ogni DC rappresenta un elemento in un nome dominio, ad esempio il nome dominio `example.ibm.com` è rappresentato come esempio, `ibm`, con utilizzando virgole per separare i valori multipli. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome componente di dominio. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientSiteDN_DNQ

Utilizzare questa proprietà per specificare un qualificativo di dominio per selezionare un certificato da inviare al server SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi qualificatore di dominio. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientSiteDN_L

Utilizzare questa proprietà per specificare un nome ubicazione per selezionare un certificato da inviare al server SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome ubicazione. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientSiteDN_O

Utilizzare questa proprietà per specificare un nome organizzazione per selezionare un certificato da inviare al server SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome organizzazione. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientSiteOU_DN

Utilizzare questa proprietà per specificare un nome OU (Organizational Unit) per selezionare un certificato da inviare al server SSL/TLS. È possibile specificare più OU separandole con virgole. (Corrisponde a una virgola letterale anteposto un carattere barra retroversa (\).) La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome OU. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate.

SSLClientSiteDN_PC

Utilizzare questa proprietà per specificare un CAP per selezionare un certificato da inviare al server SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi codice postale. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientSiteDN_ST

Utilizzare questa proprietà per specificare un nome stato per selezionare un certificato da inviare al server SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome di stato. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientSiteDN_Street

Utilizzare questa proprietà per specificare un nome strada per selezionare un certificato da inviare al server SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome strada. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato

quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientSiteDN_T

Utilizzare questa proprietà per specificare un titolo per selezionare un certificato da inviare al server SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi titolo. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLClientSiteUID_DN

Utilizzare questa proprietà per specificare un ID utente per selezionare un certificato da inviare al server SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi ID utente. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

Etichetta SSLClientSite

Utilizzare questa propriet ... per specificare un nome etichetta per selezionare un certificato da inviare al server SSL/TLS. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome etichetta. Se si modifica questa proprietà (e **SSLClient** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLExitData

Utilizzare questa proprietà per fornire una stringa definita dall'utente da passare all'uscita.

SSLExitName

Utilizzare questa proprietà per definire il nome classe per l'uscita che verrà richiamata quando l'instradamento agisce come un client SSL/TLS o un server SSL/TLS. Il nome deve includere qualsiasi nome pacchetto; ad esempio, `com.ibm.mq.ippt.exit.TestExit`.

SSLExitPath

Utilizzare questa proprietà per definire l'ubicazione dell'uscita da utilizzare per caricare una copia dell'uscita. Il nome deve essere un nome completo da utilizzare per individuare il file di classe o il nome di un file `.jar` che contiene il file di classe; ad esempio, `C:\mqipt\exits o C:\mqipt\exits\exits.jar`.

SSLExitTimeout

Utilizzare questa proprietà per definire per quanto tempo MQIPT attende il completamento dell'uscita prima di terminare la richiesta di connessione. Il valore `0` indica che MQIPT attende indefinitamente.

SSLPlainConnections

Utilizzare questa proprietà per specificare se SSL/TLS è obbligatorio per le connessioni alla porta del listener MQIPT di un instradamento configurato per accettare le connessioni SSL/TLS in entrata. Questa proprietà è applicabile agli instradamenti che hanno la proprietà **SSLServer** o **SSLProxyMode** impostata su `true`. Se abilitata, questa proprietà consente alle connessioni non codificate di connettersi alla porta del listener di instradamento, il che significa che MQIPT può inoltrare tutte le connessioni IBM MQ alla porta del listener del gestore code indipendentemente dal fatto che la connessione sia codificata. Se non imposti questo parametro o lo imposti su `false`, sono consentite solo le connessioni SSL/TLS in entrata. Se si modifica questa proprietà, l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLProxyMode

Imposta questa proprietà su `true` per fare in modo che l'instradamento accetti solo le richieste di connessione client SSL/TLS e per eseguire il tunnel della richiesta direttamente alla destinazione. Se si modifica questa proprietà, l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate. Questa proprietà non può essere utilizzata insieme alle seguenti proprietà:

- **SocksClient**

- **SocksServer**
- **SSLClient**
- **SSLServer**

Server SSL

Imposta questa proprietà su `true` per fare in modo che la rotta agisca come un server SSL/TLS e accetti le connessioni SSL/TLS in entrata. L'impostazione di **SSLServer** su `true` implica che il chiamante è un altro MQIPT che funge da client SSL/TLS o è un client IBM MQ o un gestore code con SSL/TLS abilitato.

Se si imposta **SSLServer** su `true`, è necessario specificare un keyring del server SSL/TLS utilizzando la proprietà **SSLServerKeyRing** oppure configurare MQIPT per utilizzare l'hardware crittografico impostando la proprietà **SSLServerKeyRingUseCryptoHardware**.

Se si modifica questa proprietà, l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

Questa proprietà non può essere utilizzata insieme alle seguenti proprietà:

- **SocksServer**
- **SSLProxyMode**

Anello SSLServerCAKey

Il nome file completo del file key ring contenente certificati CA, utilizzato per autenticare i certificati dal client SSL/TLS. Su piattaforme Windows, è necessario utilizzare una doppia barra rovesciata (\\) come separatore di file. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate.

SSLServerCAKeyRingPW

La parola d'ordine per aprire il file keyring CA del server SSL/TLS specificato con la proprietà **SSLServerCAKeyRing** o per connettersi all'archivio chiavi hardware crittografico se la proprietà **SSLServerCAKeyRingUseCryptoHardware** è impostata su `true`.

Il valore può essere una password che è stata codificata utilizzando il comando **mqiptPW** oppure il nome file completo del file che contiene una password codificata. Se si specifica un nome file su piattaforme Windows, è necessario utilizzare una doppia barra rovesciata (\\) come separatore di file. Si consiglia di migrare qualsiasi parola d'ordine key ring attualmente memorizzata in un file per utilizzare il metodo di protezione più recente e più sicuro, codificando nuovamente le parole d'ordine utilizzando il programma di utilità **mqiptPW**. Per ulteriori informazioni sulla codifica delle password nella configurazione di MQIPT, consultare [Codifica delle password memorizzate](#).

Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerCAKeyRingUseCryptoHardware

Specifica se l'hardware crittografico che supporta l'interfaccia PKCS #11 viene utilizzato come keystore per i certificati CA, utilizzati per autenticare i certificati dal client SSL/TLS. Se questa proprietà è impostata su `true`, **SSLServerCAKeyRing** non può essere impostata sullo stesso instradamento.

Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

L'utilizzo dell'hardware crittografico con MQIPT è una funzionalità IBM MQ Advanced. Per utilizzare questa funzionalità, il gestore code locale connesso mediante l'instradamento MQIPT deve avere anche la titolarità IBM MQ Advanced, IBM MQ Appliance, IBM MQ Advanced for z/OS o IBM MQ Advanced for z/OS VUE. L'instradamento non verrà avviato quando questa proprietà è impostata su `true` a meno che la proprietà globale **EnableAdvancedCapabilities** non sia impostata per confermare che è possibile utilizzare le capacità IBM MQ Advanced.

SSLServerAskClientAuth

Utilizzare questa proprietà per richiedere l'autenticazione client SSL/TLS dal server SSL/TLS. Il client SSL/TLS deve avere il proprio certificato da inviare al server SSL/TLS. Il certificato viene richiamato dal file di chiavi. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate.

Suite SSLServerCipher

Il nome della CipherSuite SSL/TLS da utilizzare sul lato server SSL/TLS. Può essere una o più delle CipherSuitessupportate. Se si lascia vuoto questo campo, viene utilizzato qualsiasi CipherSuite per i protocolli abilitati che sia compatibile con il certificato del server nel file di chiavi. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate.

CDN_SSLServer

Utilizzare questa proprietà per accettare i certificati ricevuti dal client SSL/TLS di questo paese. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome azienda. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerDN_CN

Utilizzare questa proprietà per accettare i certificati ricevuti dal client SSL/TLS con questo nome comune. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome comune. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

DDN_SSLServer

Utilizzare questa proprietà per accettare i certificati ricevuti dal client SSL/TLS di questo nome componente di dominio. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. È possibile specificare più DC separandoli con virgole. Ogni DC rappresenta un elemento in un nome dominio, ad esempio il nome dominio `example.ibm.com` è rappresentato come `example, ibm, com` utilizzando virgole per separare i valori multipli. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome componente di dominio. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

DNQ server SSL

Utilizzare questa proprietà per accettare i certificati ricevuti dal client SSL/TLS di questo qualificatore di dominio. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi qualificatore di dominio. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerDN_L

Utilizzare questa proprietà per accettare i certificati ricevuti dal client SSL/TLS di questa ubicazione. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi ubicazione. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerDN_O

Utilizzare questa proprietà per accettare i certificati ricevuti dal client SSL/TLS di questa organizzazione. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito.

La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi organizzazione. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

OU server SSL

Utilizzare questa proprietà per accettare i certificati ricevuti dal client SSL/TLS di questa OU (Organizational Unit). Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. È possibile specificare più OU separandole con virgole. (Corrisponde a una virgola letterale antepoendo un carattere barra rovesciata (\).) La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome OU. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate.

PC server SSL

Utilizzare questa proprietà per accettare i certificati ricevuti dal client SSL/TLS di questo codice postale. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi codice postale. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerDN_ST

Utilizzare questa proprietà per accettare i certificati ricevuti dal client SSL/TLS di questo stato. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi stato. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerDN_Via

Utilizzare questa proprietà per accettare i certificati ricevuti dal client SSL/TLS di questo nome strada. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome strada. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerDN_T

Utilizzare questa proprietà per accettare i certificati ricevuti dal client SSL/TLS di questo titolo. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi titolo. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

UID_DN_SSLServer

Utilizzare questa proprietà per accettare i certificati ricevuti dal client SSL/TLS di questo ID utente. Il nome può avere come prefisso o suffisso un asterisco (*) per estenderne l'ambito. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi ID utente. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerExit

Utilizzare questa proprietà per abilitare o disabilitare l'utilizzo di un'uscita quando l'instradamento funge da server SSL/TLS. Ciò consente di definire i dettagli di uscita nel file di configurazione senza che vengano effettivamente utilizzati.

Anello SSLServerKey

Il nome file completo del file key ring contenente il certificato server. Su piattaforme Windows, è necessario utilizzare una doppia barra rovesciata (\\) come separatore di file. Se si modifica questa

proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerKeyRingPW

La password per aprire il file keyring del server SSL/TLS specificato con la proprietà **SSLServerKeyRing** o per connettersi al keystore dell'hardware di crittografia se la proprietà **SSLServerKeyRingUseCryptoHardware** è impostata su `true`.

Il valore può essere una password che è stata codificata utilizzando il comando **mqiPTPW** oppure il nome file completo del file che contiene una password codificata. Se si specifica un nome file su piattaforme Windows, è necessario utilizzare una doppia barra rovesciata (\\) come separatore di file. Si consiglia di migrare qualsiasi parola d'ordine key ring attualmente memorizzata in un file per utilizzare il metodo di protezione più recente e più sicuro, codificando nuovamente le parole d'ordine utilizzando il programma di utilità **mqiPTPW**. Per ulteriori informazioni sulla codifica delle password nella configurazione di MQIPT, consultare [Codifica delle password memorizzate](#).

È necessario specificare **SSLServerKeyRingPW** se si imposta **SSLServer** su `true`.

Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerKeyRingUseCryptoHardware

Specifica se l'hardware crittografico che supporta l'interfaccia PKCS #11 viene utilizzato come keystore per il certificato del server, quando MQIPT funge da server SSL/TLS. Se questa proprietà è impostata su `true`, **SSLServerKeyRing** non può essere impostata sullo stesso instradamento.

Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

L'utilizzo dell'hardware crittografico con MQIPT è una funzionalità IBM MQ Advanced. Per utilizzare questa funzionalità, il gestore code locale connesso mediante l'instradamento MQIPT deve avere anche la titolarità IBM MQ Advanced, IBM MQ Appliance, IBM MQ Advanced for z/OS o IBM MQ Advanced for z/OS VUE. L'instradamento non verrà avviato quando questa proprietà è impostata su `true` a meno che la proprietà globale **EnableAdvancedCapabilities** non sia impostata per confermare che è possibile utilizzare le capacità IBM MQ Advanced.

SSLServerProtocols

Utilizzato per limitare la serie di protocolli socket sicuri abilitati utilizzati per accettare le connessioni in entrata alla porta del listener di instradamento per un instradamento quando **SSLServer** è impostato su `true`).

È possibile specificare più valori separandoli con virgole. Se non si specifica questa proprietà, TLS 1.2 e TLS 1.3 sono abilitati per impostazione predefinita. Per abilitare protocolli diversi da TLS 1.2 o TLS 1.3, devi specificare i protocolli da abilitare in questa proprietà e aggiungere anche il supporto per il protocollo in Java runtime environment seguendo la procedura in [Enabling deprecated protocols and CipherSuites](#). È possibile specificare uno o più dei seguenti valori.

Valore	Protocollo
SSLv3	SSL 3.0
TLSv1	TLS 1.0
TLSv1.1	TLS 1.1
TLSv1.2	TLS 1.2
TLSv1.3	TLS 1.3

Utilizzare la voce elencata nella colonna **Valore** nella proprietà route. La voce corrispondente nella colonna **Protocollo** è solo a scopo informativo.

SSLServerSiteDN_C

Utilizzare questa proprietà per specificare un nome paese per selezionare un certificato da inviare al client SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome paese. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerSiteDN_CN

Utilizzare questa proprietà per specificare un Nome comune per selezionare un certificato da inviare al client SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome comune. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerSiteDN_DC

Utilizzare questa proprietà per specificare un nome componente di dominio per selezionare un certificato da inviare al client SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. È possibile specificare più DC separandoli con virgole. Ogni DC rappresenta un elemento in un nome dominio, ad esempio il nome dominio `example.ibm.com` è rappresentato come `example,ibm,com` utilizzando virgole per separare i valori multipli. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome componente di dominio. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerSiteDN_DNQ

Utilizzare questa proprietà per specificare un qualificatore di dominio per selezionare un certificato da inviare al client SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi qualificatore di dominio. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerSiteDN_L

Utilizzare questa proprietà ... per specificare un nome ubicazione per selezionare un certificato da inviare al client SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome ubicazione. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerSiteDN_O

Utilizzare questa proprietà per specificare un nome organizzazione per selezionare un certificato da inviare al client SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome organizzazione. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerSiteOU_DN

Utilizzare questa proprietà per specificare un nome OU (Organizational Unit) per selezionare un certificato da inviare al client SSL/TLS. È possibile specificare più OU separandole con virgole. (Corrisponde a una virgola letterale anteposta a un carattere barra rovesciata (\).) La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome OU. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni a questo instradamento vengono arrestate.

SSLServerSiteDN_PC

Utilizzare questa proprietà per specificare un codice postale per selezionare un certificato da inviare al client SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi codice postale. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerSiteDN_ST

Utilizzare questa proprietà per specificare un nome stato per selezionare un certificato da inviare al client SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome di stato. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerSiteDN_Street

Utilizzare questa proprietà per specificare un nome strada per selezionare un certificato da inviare al client SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome strada. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerSiteDN_T

Utilizzare questa proprietà per specificare un titolo per selezionare un certificato da inviare al client SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi titolo. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

SSLServerSiteUID_DN

Utilizzare questa proprietà ... per specificare un ID utente per selezionare un certificato da inviare al client SSL/TLS. La corrispondenza del certificato non è sensibile al maiuscolo / minuscolo. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi ID utente. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

Etichetta SSLServerSite

Utilizzare questa proprietà per specificare un nome etichetta per selezionare un certificato da inviare al client SSL/TLS. Se non si specifica questa proprietà, i certificati vengono accettati con qualsiasi nome etichetta. Se si modifica questa proprietà (e **SSLServer** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento. Tutte le connessioni all'instradamento vengono arrestate.

Formato StoredCredentials

Utilizzare questa proprietà per indicare se i valori delle proprietà della password utilizzano il formato della password codificata supportato in MQIPT da IBM MQ 9.1.5. MQIPT è quasi sempre in grado di rilevare se le parole d'ordine sono specificate nel formato della parola d'ordine codificata. Questa proprietà deve essere impostata solo nello scenario improbabile in cui MQIPT non può distinguere automaticamente tra una password codificata e una password di testo semplice o un nome file.

Il valore può essere uno dei seguenti:

crittografato

Le proprietà della password contengono una password codificata nel formato supportato in MQIPT da IBM MQ 9.1.5.

COMPAT

Le proprietà della password contengono una password in testo semplice o, per le password keyring, il nome del file contenente una password codificata.

TCPKeepAlive

Impostare questa proprietà su `true` per abilitare l'invio periodico di pacchetti keep - alive TCP/IP per evitare che le connessioni su questo instradamento diventino inattive. Ciò riduce le possibilità che le connessioni MQIPT vengano interrotte da firewall o router. L'invio dei pacchetti keep - alive TCP/IP è controllato dai parametri di ottimizzazione del sistema operativo; consultare la documentazione del proprio sistema operativo per ulteriori informazioni su come ottimizzare il keep - alive. Se non si imposta questo parametro o lo si imposta su `false`, i pacchetti keep - alive non vengono inviati.

Traccia

Il livello di traccia richiesto per questo instradamento. L'abilitazione della traccia per un instradamento non abilita la traccia per altri instradamenti. Se hai bisogno di tracciare più di un instradamento, devi aggiungere la proprietà **Trace** alla sezione `[route]` di ogni instradamento di cui eseguire la traccia.

Il valore di questa proprietà può essere uno dei seguenti:

0

La traccia non è abilitata

Qualsiasi numero intero positivo

La traccia è abilitata

Il valore predefinito è 0.

Se la sezione `[route]` non include una proprietà **Trace**, viene utilizzata la proprietà **Trace** della sezione `[global]`. Per informazioni sulla traccia dei thread non associati a un instradamento, consultare **Trace** nella sezione `[global]`. Se una modifica a questa proprietà influisce su un instradamento, il nuovo valore viene utilizzato quando viene immesso il comando di aggiornamento. Tutte le connessioni utilizzano il nuovo valore immediatamente. L'instradamento non è arrestato.

Dati TraceUser

La quantità di dati utente nelle trasmissioni di rete ricevuti e inviati da questo instradamento tracciati, quando la traccia è abilitata per questo instradamento. Il valore può essere uno dei seguenti:

0

Non viene eseguita la traccia dei dati utente.

tutti

Viene eseguita la traccia di tutti i dati utente.

numberOfbyte

Viene tracciato il numero specificato di byte di dati, incluso il TSH (transmission segment header). Il valore specificato deve essere maggiore di 15.

UriName

Questa proprietà può essere utilizzata per modificare il nome dell'URI (Uniform Resource Identifier) della risorsa quando si utilizza un proxy HTTP, sebbene il valore predefinito sia sufficiente per la maggior parte delle configurazioni:

```
HTTP://destination:destination_port/mqipt
```

Se si modifica questa proprietà (e **HTTP** è impostata su `true`), l'instradamento viene arrestato e riavviato quando viene immesso un comando di aggiornamento.

Proprietà di mqiptAdmin

Il comando **mqiptAdmin** legge le proprietà di configurazione da un file delle proprietà specificato quando il comando viene avviato.

Le proprietà riportate di seguito possono essere specificate nel file delle proprietà utilizzato dal comando **mqiptAdmin**. I nomi proprietà sono sensibili al maiuscolo e al minuscolo.

PasswordProtectionKeyFile

Il nome del file contenente la chiave di codifica utilizzata per codificare la password del truststore specificata nella proprietà **SSLClientCAKeyRingPW** . Se questa proprietà non viene specificata, la chiave di codifica predefinita viene utilizzata per decodificare la parola d'ordine. La chiave di codifica utilizzata per codificare la password del truststore **mqiPTAdmin** può essere diversa dalla chiave di codifica utilizzata per codificare le password nel file di configurazione `mqiPT.conf` .

Anello SSLClientCAKey

Il nome file del truststore PKCS#12 da utilizzare per connessioni alla porta comandi TLS MQIPT . Il truststore deve contenere il certificato CA della CA che ha firmato il certificato server che la porta comandi TLS MQIPT è configurata per utilizzare. I caratteri barra retroversa (\) nel nome file devono essere preceduti da un carattere escape e specificati come una doppia barra retroversa (\\).

SSLClientCAKeyRingPW

La password codificata per accedere al truststore specificato utilizzando la proprietà **SSLClientCAKeyRing** . La password deve essere codificata utilizzando il comando **mqiPTPW** e il valore di questa proprietà deve essere impostato sull'output della stringa da **mqiPTPW**.

Informazioni particolari

Queste informazioni sono state sviluppate per prodotti e servizi offerti negli Stati Uniti.

IBM potrebbe non offrire i prodotti, i servizi o le funzioni descritti in questo documento in altri paesi. Consultare il rappresentante IBM locale per informazioni sui prodotti e sui servizi disponibili nel proprio paese. Ogni riferimento relativo a prodotti, programmi o servizi IBM non implica che solo quei prodotti, programmi o servizi IBM possano essere utilizzati. In sostituzione a quelli forniti da IBM possono essere usati prodotti, programmi o servizi funzionalmente equivalenti che non comportino la violazione dei diritti di proprietà intellettuale o di altri diritti dell'IBM. Tuttavia, è responsabilità dell'utente valutare e verificare il funzionamento di qualsiasi prodotto, programma o servizio non IBM.

IBM potrebbe disporre di applicazioni di brevetti o brevetti in corso relativi all'argomento descritto in questo documento. La fornitura di tale documento non concede alcuna licenza a tali brevetti. Chi desiderasse ricevere informazioni relative a licenze può rivolgersi per iscritto a:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Per richieste di licenze relative ad informazioni double-byte (DBCS), contattare il Dipartimento di Proprietà Intellettuale IBM nel proprio paese o inviare richieste per iscritto a:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

Il seguente paragrafo non si applica al Regno Unito o a qualunque altro paese in cui tali dichiarazioni sono incompatibili con le norme locali: INTERNATIONAL BUSINESS MACHINES CORPORATION FORNISCE LA PRESENTE PUBBLICAZIONE "NELLO STATO IN CUI SI TROVA" SENZA GARANZIE DI ALCUN TIPO, ESPRESSE O IMPLICITE, IVI INCLUSE, A TITOLO DI ESEMPIO, GARANZIE IMPLICITE DI NON VIOLAZIONE, DI COMMERCIALIZZABILITÀ E DI IDONEITÀ PER UNO SCOPO PARTICOLARE. Alcuni stati non consentono la rinuncia a garanzie esplicite o implicite in determinate transazioni; quindi la presente dichiarazione potrebbe non essere applicabile.

Questa pubblicazione potrebbe contenere imprecisioni tecniche o errori tipografici. Le informazioni incluse in questo documento vengono modificate su base periodica; tali modifiche vengono incorporate nelle nuove edizioni della pubblicazione. IBM si riserva il diritto di apportare miglioramenti o modifiche al prodotto/i e/o al programma/i descritti nella pubblicazione in qualsiasi momento e senza preavviso.

Tutti i riferimenti a siti Web non dell'IBM contenuti in questo documento sono forniti solo per consultazione e non rappresenta in alcun modo un'approvazione di tali siti. I materiali reperibili in tali siti Web non fanno parte dei materiali relativi a questo prodotto IBM e l'utilizzo di tali siti è responsabilità dell'utente.

Tutti i commenti e i suggerimenti inviati potranno essere utilizzati liberamente da IBM e diventeranno esclusiva della stessa.

Coloro che detengono la licenza su questo programma e desiderano avere informazioni su di esso allo scopo di consentire (i) uno scambio di informazioni tra programmi indipendenti ed altri (compreso questo) e (ii) l'uso reciproco di tali informazioni, dovrebbero rivolgersi a:

IBM Corporation
Coordinatore interoperabilità software, Dipartimento 49XA
Autostrada 3605 52 N

Rochester, MN 55901
U.S.A.

Queste informazioni possono essere rese disponibili secondo condizioni contrattuali appropriate, compreso, in alcuni casi, il pagamento di un addebito.

Il programma su licenza descritto in queste informazioni e tutto il materiale su licenza disponibile per esso sono forniti da IBM in base ai termini dell' IBM Customer Agreement, IBM International Program License Agreement o qualsiasi altro accordo equivalente tra le parti.

Tutti i dati relativi alle prestazioni contenuti in questo documento sono stati determinati in un ambiente controllato. Pertanto, i risultati ottenuti in altri ambienti operativi possono variare in modo significativo. Alcune misurazioni potrebbero essere state fatte su sistemi a livello di sviluppo e non vi è alcuna garanzia che queste misurazioni saranno le stesse sui sistemi generalmente disponibili. Inoltre, alcune misurazioni potrebbero essere state stimate mediante estrapolazione. I risultati quindi possono variare. Gli utenti di questo documento dovrebbero verificare i dati applicabili per il loro ambiente specifico.

Le informazioni relative a prodotti non IBM provengono dai fornitori di tali prodotti, dagli annunci pubblicati o da altre fonti pubblicamente disponibili. IBM non ha verificato tali prodotti e, pertanto, non può garantirne l'accuratezza delle prestazioni. Eventuali commenti relativi alle prestazioni dei prodotti non IBM devono essere indirizzati ai fornitori di tali prodotti.

Tutte le dichiarazioni riguardanti la direzione o l'intento futuro di IBM sono soggette a modifica o ritiro senza preavviso e rappresentano solo scopi e obiettivi.

Questa pubblicazione contiene esempi di dati e prospetti utilizzati quotidianamente nelle operazioni aziendali. Per poterli illustrare nel modo più completo possibile, gli esempi riportano nomi di persone, società, marchi e prodotti. Tutti questi nomi sono fittizi e qualsiasi somiglianza con nomi ed indirizzi adoperati da imprese realmente esistenti sono una mera coincidenza.

LICENZA SUL COPYRIGHT:

Queste informazioni contengono programmi applicativi di esempio in lingua originale, che illustrano le tecniche di programmazione su diverse piattaforme operative. È possibile copiare, modificare e distribuire questi programmi di esempio sotto qualsiasi forma senza alcun pagamento alla IBM, allo scopo di sviluppare, utilizzare, commercializzare o distribuire i programmi applicativi in conformità alle API (application programming interface) a seconda della piattaforma operativa per cui i programmi di esempio sono stati scritti. Questi esempi non sono stati testati approfonditamente tenendo conto di tutte le condizioni possibili. IBM, quindi, non può garantire o sottintendere l'affidabilità, l'utilità o il funzionamento di questi programmi.

Se si sta visualizzando queste informazioni in formato elettronico, le fotografie e le illustrazioni a colori potrebbero non apparire.

Informazioni sull'interfaccia di programmazione

Le informazioni sull'interfaccia di programmazione, se fornite, consentono di creare software applicativo da utilizzare con questo programma.

Questo manuale contiene informazioni sulle interfacce di programmazione che consentono al cliente di scrivere programmi per ottenere i servizi di IBM MQ.

Queste informazioni, tuttavia, possono contenere diagnosi, modifica e regolazione delle informazioni. La diagnosi, la modifica e la regolazione delle informazioni vengono fornite per consentire il debug del software applicativo.

Importante: Non utilizzare queste informazioni di diagnosi, modifica e ottimizzazione come interfaccia di programmazione poiché sono soggette a modifica.

Marchi

IBM, il logo IBM, ibm.com, sono marchi di IBM Corporation, registrati in molte giurisdizioni nel mondo. Un elenco aggiornato dei marchi IBM è disponibile sul web in "Copyright and trademark

information"www.ibm.com/legal/copytrade.shtml. Altri nomi di prodotti e servizi potrebbero essere marchi di IBM o altre società.

Microsoft e Windows sono marchi di Microsoft Corporation negli Stati Uniti, in altri paesi o entrambi.

UNIX è un marchio registrato di The Open Group negli Stati Uniti e/o in altri paesi.

Linux è un marchio registrato di Linus Torvalds negli Stati Uniti e/o in altri paesi.

Questo prodotto include il software sviluppato da Eclipse Project (<https://www.eclipse.org/>).

Java e tutti i marchi e i logo Java sono marchi registrati di Oracle e/o di società affiliate.



Numero parte:

(1P) P/N: