

9.1

*Administración de IBM MQ*

**IBM**

**Nota**

Antes de utilizar esta información y el producto al que se refiere, lea la información en [“Avisos” en la página 595](#).

Esta edición se aplica a la versión 9 release 1 de IBM® MQ y a todos los releases y modificaciones posteriores hasta que se indique lo contrario en nuevas ediciones.

Cuando envía información a IBM, otorga a IBM un derecho no exclusivo para utilizar o distribuir la información de la forma que considere adecuada, sin incurrir por ello en ninguna obligación con el remitente.

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

---

# Contenido

<b>Administración de.....</b>	<b>7</b>
Formas de administrar los gestores de colas de IBM MQ y los recursos asociados.....	8
Administración de IBM MQ mediante mandatos de control.....	11
Administración utilizando mandatos MQSC.....	12
Ejecución interactiva de mandatos MQSC.....	14
Ejecución de mandatos MQSC desde archivos de texto.....	17
Orígenes desde los que puede emitir mandatos MQSC en z/OS.....	20
Configuración automática de un script de mandato de script de WebSphere MQ en el inicio.....	21
Resolución de problemas con mandatos MQSC.....	22
Valores y caracteres genéricos con significados especiales.....	23
Automatización de la administración de IBM MQ utilizando mandatos PCF.....	24
Introducción a los Formatos de mandato programable de IBM MQ.....	25
Utilizar la MQAI para simplificar el uso de los PCF.....	37
Administración utilizando REST API.....	74
Iniciación a administrative REST API.....	75
Administración remota mediante REST API.....	80
Indicaciones de fecha y hora de REST API.....	84
Manejo de errores de REST API.....	84
Descubrimiento de REST API.....	87
Soporte multilingüístico de REST API.....	88
Versiones de REST API.....	90
Administración utilizando una consola web.....	91
Cómo empezar con la consola web.....	92
Visita rápida de la New Web Console.....	93
Conmutación de tipos de consola en IBM MQ for Multiplatforms.....	97
Dashboard Web Console.....	97
Administración utilizando IBM MQ Explorer.....	133
Lo que se puede hacer con IBM MQ Explorer.....	134
Configurar IBM MQ Explorer.....	135
Utilización de la aplicación Barra de tareas de IBM MQ (sólo Windows).....	141
La aplicación de supervisor de alertas de IBM MQ (sólo Windows).....	141
Trabajar con objetos de IBM MQ locales.....	142
Trabajar con gestores de colas.....	142
Detención de canales MQI.....	152
Trabajar con colas locales.....	153
Trabajar con colas remotas.....	162
Trabajar con colas alias.....	165
Trabajar con colas modelo.....	167
Trabajar con colas de mensajes no entregados.....	168
Trabajar con temas administrativos.....	187
Trabajar con suscripciones.....	190
Trabajar con servicios.....	194
Gestión de objetos para desencadenamiento.....	202
Utilización del programa de utilidad <b>dmpmqmsg</b> entre dos sistemas.....	204
Trabajar con objetos de IBM MQ remotos.....	207
Configuración de gestores de colas para la administración remota.....	208
Gestión del servidor de mandatos para la administración remota.....	212
Emisión de mandatos MQSC en un gestor de colas remoto.....	213
Conversión de datos entre juegos de caracteres codificados.....	215
Administración de Managed File Transfer.....	220
Iniciar un agente de MFT.....	220
Listar agentes de MFT.....	225

Detener un agente de MFT.....	225
Inicio de una nueva transferencia de archivos.....	227
Creación de una transferencia planificada de archivos.....	230
Cómo trabajar con transferencias de archivos pendientes.....	231
Desencadenamiento de una transferencia de archivos.....	232
Supervisión de transferencias de archivos que están en curso.....	233
Visualización del estado de transferencias de archivos en el Registro de transferencias.....	235
Supervisión de recursos de MFT.....	238
Cómo trabajar con plantillas de transferencia de archivos.....	266
Transferencia de datos de archivos a mensajes.....	269
Transferir datos de mensajes a archivos.....	277
El puente de protocolo.....	283
El puente Connect:Direct.....	299
Cómo trabajar con MFT desde IBM Integration Bus.....	314
Recuperación y reinicio de MFT.....	314
Establecimiento de un tiempo de espera para la recuperación de transferencias estancadas.....	315
Administración de MQ Telemetry.....	321
Configurar un gestor de colas para telemetría en Linux y AIX.....	321
Configurar un gestor de colas para telemetría en Windows.....	323
Configuración de las colas distribuidas para enviar mensajes a clientes MQTT.....	325
Identificación, autorización y autenticación de clientes MQTT.....	327
Autenticación de canal de telemetría mediante TLS.....	334
Privacidad de las publicaciones en los canales de telemetría.....	335
Configuración TLS de clientes y canales de telemetría de MQTT Java.....	336
Configuración JAAS del canal de telemetría.....	341
Administración de IBM MQ Light.....	343
Visualización de objetos de IBM MQ que utilizan clientes MQ Light.....	343
Identificación, autorización y autenticación de clientes MQ Light.....	344
Privacidad de las publicaciones en los canales.....	347
Configuración de clientes MQ Light con TLS.....	347
Desconexión de clientes MQ Light del gestor de colas.....	348
Administración de multidifusión.....	349
Iniciación a la multidifusión.....	349
Topología de temas de IBM MQ Multicast.....	350
Control del tamaño de mensajes de multidifusión.....	351
Habilitación de la conversión de datos para la mensajería de Multicast.....	353
Supervisión de aplicaciones de multidifusión.....	353
Fiabilidad de los mensajes de multidifusión.....	354
Tareas avanzadas de multidifusión.....	355
Administración de IBM MQ for IBM i.....	358
Gestión de IBM MQ for IBM i utilizando mandatos CL.....	359
Formas alternativas de administrar IBM MQ for IBM i.....	372
Gestión de trabajo en IBM i.....	378
Disponibilidad, copia de seguridad, recuperación y reinicio en IBM i.....	385
Progresivo IBM MQ for IBM i.....	430
Administración de IBM MQ for z/OS.....	434
Emisión de mandatos a IBM MQ for z/OS.....	434
Los programas de utilidad de IBM MQ for z/OS.....	443
Operando IBM MQ for z/OS.....	446
Escribir programas para administrar IBM MQ for z/OS.....	467
Gestión de recursos de IBM MQ en z/OS.....	479
Recuperación y reinicio en z/OS.....	520
IBM MQ y IMS.....	543
Operación de Advanced Message Security en z/OS.....	556
Proveedor de servicios IBM MQ for z/OS para IBM z/OS Connect EE.....	557
Administración y configuración de IBM MQ Internet Pass-Thru.....	582
Iniciar y detener MQIPT.....	583
administrar MQIPT utilizando la línea de mandatos.....	587

Utilización de IPT Administration Client.....	588
Realización de copias de seguridad.....	591
Ajuste de rendimiento.....	592
<b>Avisos.....</b>	<b>595</b>
Información acerca de las interfaces de programación.....	596
Marcas registradas.....	597



# Administración de IBM MQ

Para administrar los gestores de colas de IBM MQ y los recursos asociados, elija el método que prefiera de un conjunto de tareas que puede utilizar para activar y gestionar estos recursos.

## Acerca de esta tarea

Puede administrar los objetos de IBM MQ de forma local o remota:

### Administración local

La administración local es aquella en la que las tareas administrativas se realizan en cualquier gestor de colas definido en el sistema local. Puede acceder a otros sistemas, por ejemplo mediante el programa de emulación de terminal **telnet** de TCP/IP, y llevar a cabo la administración allí. En IBM MQ, esto puede considerarse como administración local porque no hay ningún canal implicado, es decir, la comunicación la gestiona el sistema operativo.

Para obtener más información, consulte [“Trabajar con objetos de IBM MQ locales”](#) en la página 142.

### Administración remota



IBM MQ soporta la administración desde un único punto de contacto a través de una administración remota. La administración remota permite ejecutar mandatos desde el sistema local que se procesan en otro sistema y también se aplica a IBM MQ Explorer. Por ejemplo, puede emitir un mandato remoto para cambiar una definición de cola en un gestor de colas remoto. No es necesario que se conecte a ese sistema, aunque sí que necesita tener definidos los canales apropiados. El gestor de colas y el servidor de mandatos del sistema de destino deben estar ejecutándose.

Algunos mandatos no se pueden emitir de esta manera, especialmente los que crean o inician gestores de colas y los que inician servidores de mandatos. Para realizar este tipo de tarea, hay que iniciar la sesión en el sistema remoto y ejecutar los mandatos desde allí o crear un proceso que pueda ejecutarlos automáticamente. Esta restricción también se aplica a IBM MQ Explorer.






Para obtener más información, consulte [“Trabajar con objetos de IBM MQ remotos”](#) en la página 207.


Existe una serie de métodos diferentes que puede utilizar para crear y administrar los gestores de colas y sus recursos relacionados en IBM MQ. Estos métodos incluyen las interfaces de línea de mandatos, las interfaces gráficas de usuario y una API de administración.

Existen distintos conjuntos de mandatos que puede utilizar para administrar IBM MQ en función de la plataforma:

- [“Mandatos de control de IBM MQ”](#) en la página 8
- [“mandatos de script de IBM MQ \(MQSC\)”](#) en la página 8
- [“Formatos de mandato programable \(PCF\)”](#) en la página 9
-  [La administrative REST API](#)
-  [“Lenguaje de control \(CL\) de IBM i”](#) en la página 9

También hay las otras opciones siguientes para crear y gestionar los objetos de IBM MQ:

-   [“IBM MQ Explorer”](#) en la página 10
-  [“IBM MQ Console”](#) en la página 10
-  [“Con la aplicación de configuración predeterminada de Windows”](#) en la página 10
-  [“Microsoft Cluster Service \(MSCS\)”](#) en la página 10

 Si desea más información sobre las interfaces y las opciones de administración en IBM MQ for z/OS, consulte [“Administración de IBM MQ for z/OS”](#) en la página 434.

Puede automatizar algunas tareas de administración y supervisión para ambos gestores de colas, locales y remotos, mediante mandatos PCF. Estos mandatos también se pueden simplificar utilizando la Interfaz de administración de IBM MQ (MQAI) en algunas plataformas. Si desea más información sobre cómo automatizar las tareas de administración, consulte [“Automatización de la administración de IBM MQ utilizando mandatos PCF”](#) en la página 24.

### Conceptos relacionados

[Visión general técnica de IBM MQ](#)

### Tareas relacionadas

[Planificación](#)

[Configuración](#)

### Referencia relacionada

[Comparativa de conjuntos de mandatos](#)

## Formas de administrar los gestores de colas de IBM MQ y los recursos asociados

---

Existen varias opciones diferentes para administrar los gestores de colas de IBM MQ y los recursos asociados.

### Mandatos de control de IBM MQ



Puede realizar tareas administrativas en los propios gestores de colas mediante los mandatos de control.

IBM MQ for Windows y los sistemas UNIX and Linux<sup>®</sup> proporcionan los *mandatos de control* que se emiten en la línea de mandatos del sistema.

Los mandatos de control se describen en [Creación y gestión de gestores de colas en Multiplatforms](#). Para la referencia de mandato de los mandatos de control, consulte [Mandatos de control de IBM MQ](#).

### mandatos de script de IBM MQ (MQSC)

Utilice mandatos MQSC para gestionar objetos de gestor de colas, incluido el propio gestor de colas, colas, definiciones de proceso, canales, canales de conexión de cliente, escuchas, servicios, y objetos de información de autenticación.

Para emitir mandatos MQSC a un gestor de colas se utiliza el mandato **runmqsc**. Esto puede hacerse interactivamente, entrando los mandatos en el teclado, o puede redireccionarse el dispositivo de entrada estándar (stdin) para que ejecute una secuencia de mandatos de un archivo de texto ASCII. En ambos casos, el formato de los mandatos es el mismo.

Puede ejecutar el mandato **runmqsc** en tres modalidades, según los indicadores definidos en el mandato:

- *Modalidad de verificación*, donde los mandatos MQSC se verifican en un gestor de colas local, pero no se ejecutan.
- *Modalidad directa*, en la que los mandatos MQSC se ejecutan en un gestor de colas local.
- *Modalidad indirecta*, en la que los mandatos MQSC se ejecutan en un gestor de colas remoto.

Los mandatos MQSC están disponibles en todas las plataformas, incluyendo IBM i y z/OS. Los mandatos MQSC se resumen en la sección [Comparación de conjuntos de mandatos](#).



En UNIX, Linux, and Windows, puede utilizar el mandato MQSC como mandato individual emitido en la línea de mandatos del sistema. Para emitir varios mandatos, o mandatos más complicados, el mandato MQSC se puede crear en un archivo que se ejecute desde la línea de mandatos. Los mandatos MQSC se pueden enviar a un gestor de colas remoto. Si desea detalles completos, consulte [“Ejecución de mandatos MQSC desde archivos de texto”](#) en la página 17.



**IBM i** Para emitir los mandatos en un servidor IBM i, cree una lista de mandatos en un archivo de script y a continuación ejecute el archivo mediante el mandato STRMQMMQSC.

**Notas:** **IBM i**

1. No utilice la biblioteca QTEMP como biblioteca de entrada para STRMQMMQSC, ya que el uso de la biblioteca QTEMP es limitado. Debe utilizar otra biblioteca como un archivo de entrada para el mandato.
2. En IBM i, las respuestas de MQSC a los mandatos emitidos desde un archivo de script se devuelven en un archivo spool.

Para obtener más información sobre la utilización de los mandatos MQSC, consulte [“Administración utilizando mandatos MQSC”](#) en la página 12.

## Formatos de mandato programable (PCF)

Los formatos de mandato programable (PCF) definen mensajes de mandato y de respuesta que se pueden intercambiar entre un programa y cualquier gestor de colas (que admita PCF) en una red. Puede utilizar los mandatos PCF en el programa de aplicación de gestión de sistemas para la administración de objetos de IBM MQ: los objetos de información de autenticación, los canales, los escuchas de canales, las listas de nombres, las definiciones de proceso, los gestores de colas, las colas, los servicios y las clases de almacenamiento. La aplicación puede operar desde un único punto de la red para comunicar información de mandato y de respuesta a cualquier gestor de colas, local o remoto, utilizando el gestor de colas local.

Si desea más información sobre los PCF, consulte [“Introducción a los Formatos de mandato programable de IBM MQ”](#) en la página 25.

Para la definición de los PCF y las estructuras para los mandatos y las respuestas, consulte [Referencia de formatos de mandato programable](#).

## administrative REST API

**V 9.1.0**

administrative REST API proporciona una interfaz RESTful que puede utilizar para administrar IBM MQ. Cuando utiliza administrative REST API, se invocan métodos HTTP en un URL que representa un objeto de IBM MQ. Por ejemplo, puede solicitar información sobre las instalaciones de IBM MQ utilizando el método HTTP GET en el URL siguiente:

**V 9.1.0**

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

Puede utilizar administrative REST API con la implementación de HTTP/REST de un lenguaje de programación o utilizando herramientas como cURL o un complemento de navegador de cliente REST.

Si desea más información, consulte [administrative REST API](#)

## Lenguaje de control (CL) de IBM i

**IBM i**

Este lenguaje se puede utilizar para emitir mandatos de administración a IBM MQ for IBM i. Los mandatos se pueden emitir en la línea de mandatos o bien escribiendo un programa de CL. Estos mandatos realizan funciones similares a los mandatos PCF, pero el formato es diferente. Los mandatos CL se han diseñado de forma exclusiva para los servidores y las respuestas CL son legibles para las personas, mientras que los mandatos PCF son independientes de la plataforma y ambos formatos, de mandato y de respuesta, están pensados para el uso del programa.

Para obtener información detallada del lenguaje de control (CL) de IBM i, consulte [Mandatos CL de IBM MQ for IBM i](#).

## IBM MQ Explorer

Windows Linux

Utilizando IBM MQ Explorer, puede realizar las acciones siguientes:

- Definir y controlar diversos recursos como, por ejemplo, gestores de colas, colas, definiciones de proceso, listas de nombres, canales, canales de conexión de cliente, escuchas, servicios y clústeres.
- Iniciar o detener un gestor de colas local y los procesos asociados al mismo.
- Ver gestores de colas y sus objetos asociados en su estación de trabajo o desde otras estaciones de trabajo.
- Comprobar el estado de gestores de colas, clústeres y canales.
- Comprobar qué aplicaciones, usuarios o canales tienen una cola determinada abierta, a partir del estado de la cola.

En sistemas Windows y Linux, puede iniciar IBM MQ Explorer utilizando el menú del sistema, el archivo ejecutable MQExplorer o el mandato **strmqcfcg**.

Linux

En Linux, para iniciar correctamente IBM MQ Explorer, debe poder grabar un archivo en el directorio de inicio y debe existir un directorio de inicio.

Para obtener más información, consulte [“Administración utilizando IBM MQ Explorer”](#) en la página 133.

Puede utilizar IBM MQ Explorer para administrar gestores de colas remotos en otras plataformas, incluyendo z/OS.

IBM MQ Explorer se puede instalar como parte de la instalación del producto (consulte [Instalación y desinstalación de IBM MQ](#)) o desde la descarga de IBM MQ Explorer autónoma disponible en Fix Central (consulte [Instalación y desinstalación de IBM MQ Explorer como una aplicación autónoma en Linux y Windows](#)).

## IBM MQ Console

V 9.1.0

Puede utilizar IBM MQ Console para administrar IBM MQ desde un navegador web.

Para obtener más información, consulte [“Administración utilizando una consola web”](#) en la página 91.

## Con la aplicación de configuración predeterminada de Windows

Windows

Puede utilizar el programa de configuración predeterminada de Windows para crear un conjunto *inicial* (o predeterminado) de objetos IBM MQ. En la [Tabla 1: Objetos creados por la aplicación de configuración predeterminada de Windows](#) se muestra un resumen de los objetos predeterminados que se crean.

## Microsoft Cluster Service (MSCS)

Windows

Microsoft Cluster Service (MSCS) le permite conectar servidores a un *clúster*, lo que proporciona una mayor disponibilidad de datos y aplicaciones y facilita la gestión del sistema. MSCS puede detectar y recuperarse automáticamente de los errores del servidor o de las aplicaciones.

Es importante no confundir los clústeres en el sentido de MSCS con los clústeres de IBM MQ. La diferencia es la siguiente:

### Clústeres de IBM MQ

Son grupos de dos o más gestores de colas en uno o varios sistemas que proporcionan una interconexión automática y permiten que se compartan las colas entre los mismos para fines de equilibrio de carga y redundancia.

## Clústeres de MSCS

Son Grupos de sistemas, conectados entre sí y configurados de tal modo que si se produce un error en uno de ellos, MSCS ejecuta una *sustitución por anomalía*, transfiere los datos de estado de las aplicaciones del sistema anómalo a otro sistema del clúster y reinicia su ejecución en el mismo.

El tema [Soporte de Microsoft Cluster Service \(MSCS\)](#) proporciona información detallada sobre cómo configurar el sistema IBM MQ for Windows para utilizar MSCS.

ULW

## Administración de IBM MQ mediante mandatos de control

Los mandatos de control proporcionan una manera de realizar una serie de tareas de administración de IBM MQ en UNIX, Linux, and Windows.

Si desea emitir mandatos de control, su ID de usuario debe ser miembro del grupo mqm para la mayoría de los mandatos de control. Para obtener más información, consulte [Autorización para administrar IBM MQ en UNIX, Linux, and Windows](#). Además, tenga en cuenta la información específica del entorno, para la plataforma o plataformas que utilice su empresa.

Al utilizar los mandatos de control que funcionan en un gestor de colas, debe utilizar el mandato desde la instalación asociada con el gestor de colas con el que está trabajando.

Cuando se utilizan mandatos de control que funcionan en un gestor de colas configurado para utilizar la autenticación de conexiones con CHCKLOCL(REQUIRED) y se observa un error de conexión, puede:

- Proporcionar un ID de usuario y una contraseña si el mandato de control lo permite.
- Utilizar equivalentes MQSC de los mandatos de control allí donde existan.
- Iniciar el gestor de colas utilizando la opción `-ns`, cuando deben ejecutarse mandatos de control que no pueden conectarse.

Para obtener una lista completa de los mandatos de control, consulte [Los mandatos de control de IBM MQ](#).

## Utilización de mandatos de control en sistemas Windows

Windows

En IBM MQ for Windows, especifique los mandatos de control en un indicador de mandatos.

Los mandatos de control y sus distintivos no distinguen entre mayúsculas y minúsculas, pero los argumentos en dichos mandatos como, por ejemplo, nombres de cola y nombres de gestor de colas, sí distinguen entre mayúsculas y minúsculas.

Por ejemplo, en el mandato:

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

- El nombre del mandato puede entrarse en mayúsculas o minúsculas o en una combinación de ambas. Todos ellos son válidos: `crtmqm`, `CRTMQM` y `CRTmqm`.
- El indicador puede entrarse como `-u`, `-U`, `/u` o `/U`.
- `SYSTEM.DEAD.LETTER.QUEUE` y `jupiter.queue.manager` deben especificarse exactamente como se muestra.

## Utilización de mandatos de control en sistemas UNIX y Linux

Linux

UNIX

En los sistemas IBM MQ for UNIX y Linux, los mandatos de control se especifican en una ventana de shell.

En los entornos UNIX, los mandatos de control, incluido el nombre del mandato propiamente dicho, los indicadores y los argumentos que pueda haber distinguen entre mayúsculas y minúsculas. Por ejemplo, en el mandato:

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

- El nombre del mandato debe ser `crtmqm`, no `CRTMQM`.
- El indicador debe ser `-u`, no `-U`.
- La cola de mensajes no entregados es `SYSTEM.DEAD.LETTER.QUEUE`.
- El argumento se ha especificado como `jupiter.queue.manager`, lo que es distinto de `JUPITER.queue.manager`.

Preste atención y escriba los mandatos exactamente igual que en los ejemplos.

### Referencia relacionada


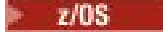
[Referencia de los mandatos de control de IBM MQ](#)

## Administración utilizando mandatos MQSC

Puede utilizar mandatos MQSC para gestionar objetos de gestor de colas, incluido el propio gestor de colas, colas, definiciones de proceso, canales, canales de conexión de cliente, escuchas, servicios, listas de nombres, clústeres y objetos de información de autenticación. Los mandatos MQSC están disponibles en todas las plataformas.

### Acerca de esta tarea


El modo en que se emiten mandatos MQSC depende de la plataforma.

-  En UNIX, Linux, and Windows, emite mandatos MQSC a un gestor de colas utilizando el mandato `runmqsc`. Puede ejecutar el mandato `runmqsc` de una serie de maneras:
  - De forma interactiva, emitiendo mandatos desde un teclado. Consulte [“Ejecución interactiva de mandatos MQSC”](#) en la página 14.
  - Desde un archivo de texto ASCII. Consulte [“Ejecución de mandatos MQSC desde archivos de texto”](#) en la página 17.
  - En un gestor de colas remoto. Consulte [“Trabajar con objetos de IBM MQ remotos”](#) en la página 207.
-  En z/OS, los mandatos MQSC se pueden emitir desde una serie de orígenes, en función del mandato. Para obtener más información, consulte [“Orígenes desde los que puede emitir mandatos MQSC en z/OS”](#) en la página 20.

Los mandatos MQSC se detallan en la sección [Mandatos MQSC](#).

### Procedimiento

- Cada mandato empieza por un parámetro primario (un verbo) y a éste le sigue un parámetro secundario (un sustantivo). Después le sigue el nombre o nombre genérico del objeto (entre paréntesis) si es que lo hay, como suele ocurrir en la mayoría de los mandatos. A continuación, los parámetros pueden aparecer, normalmente, en cualquier orden; si un parámetro tiene un valor correspondiente, éste debe aparecer directamente después del parámetro con el que está relacionado.

**Nota:**  En z/OS, no es necesario que el parámetro secundario sea el segundo.

- Las palabras clave, los paréntesis y los valores pueden estar separados por el número de blancos y de comas que se desee. Una coma mostrada en los diagramas de sintaxis siempre puede ser reemplazada por uno o más espacios en blanco. Debe haber como mínimo un blanco inmediatamente antes de cada parámetro (después del parámetro primario) excepto en z/OS.

- Al principio o final de un mandato puede aparecer un número cualquiera de espacios en blanco, así como entre parámetros, signos de puntuación y valores. Por ejemplo, el mandato siguiente es válido:

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

Los blancos dentro de un par de comillas son significativos.

- Las comas adicionales pueden aparecer en cualquier lugar donde estén permitidos los espacios en blanco y se tratan como si fueran espacios en blanco (a menos, por supuesto, que estén dentro de series entre comillas).
- Los parámetros repetidos no están permitidos. Repetir un parámetro con su versión "NO", como en REPLACE NOREPLACE, tampoco está permitido.
- Las series que contienen espacios en blanco, caracteres en minúsculas o caracteres especiales deben ir entre comillas simples, a menos que sea verdad una de las afirmaciones siguientes:

- Los caracteres especiales son uno o varios de los caracteres siguientes:

- Punto (.)
- Barra inclinada (/)
- Subrayado (\_)
- Signo de porcentaje (%)


-  El mandato se emite desde los paneles de control y operaciones de IBM MQ for z/OS.

- La serie es un valor genérico que acaba con un asterisco. (en IBM i estos se deben especificar entre comillas simples)


- La serie es un solo asterisco, por ejemplo, TRACE(\*) (en IBM i estas se deben especificar entre comillas simples)

- La serie es una especificación de rango que contiene dos puntos, por ejemplo, CLASS(01:03)

Si la cadena propiamente dicha contiene comillas simples, las comillas simples se representan con dos comillas simples. Los caracteres en minúsculas que no estén entre comillas se convertirán en mayúsculas.

-  En Multiplatforms, una serie que no contiene caracteres (es decir, dos comillas simples sin espacio entre medias) se interpreta como un espacio en blanco entre comillas simples, es decir, se interpreta del mismo modo que ("). La excepción a esto es si el atributo que se utiliza es uno de los atributos siguientes, cuando dos comillas simples sin espacio se interpretan como una serie de longitud cero:

- TOPICSTR
- SUB
- USERDATA
- SELECTOR

-  En z/OS, si desea un espacio en blanco entre comillas simples, debe especificarlo como tal ("). Una serie que no contiene caracteres (") es lo mismo que especificar ().

- Los espacios en blanco al final en atributos de serie que se basan en tipos MQCHARV como, por ejemplo, SELECTOR, los datos de subusuario, son significativos, es decir, 'abc ' no es igual que 'abc'.

- Un signo de abrir paréntesis seguido de un signo de cerrar paréntesis, sin ninguna información significativa entre ellos, no es válido, excepto cuando se indique de forma específica. Por ejemplo, la serie siguiente no es válida:

```
NAME ( )
```

- Las palabras clave no son sensibles a las mayúsculas y minúsculas: AltER, alter y ALTER son todas correctas. Lo que no esté entre comillas se convertirá en mayúsculas.
- Se han definido sinónimos para algunos parámetros. Por ejemplo, DEF siempre es un sinónimo de DEFINE; por lo tanto, DEF QLOCAL es válido. Sin embargo, los sinónimos no son simplemente palabras abreviadas; DEFI no es un sinónimo de DEFINE.

**Nota:** No hay ningún sinónimo para el parámetro DELETE. Esto es así para evitar borrar accidentalmente objetos al utilizar DEF, el sinónimo de DEFINE.

- Los mandatos MQSC utilizan determinados caracteres especiales que tienen determinados significados. Para obtener más información sobre estos caracteres especiales y cómo utilizarlos, consulte [“Valores y caracteres genéricos con significados especiales”](#) en la página 23.

### Referencia relacionada


[runmqsc \(ejecutar mandatos MQSC\)](#)

## Ejecución interactiva de mandatos MQSC

Puede utilizar los mandatos MQSC de forma interactiva, mediante una ventana de mandatos o shell.

### Antes de empezar

Puede establecer la solicitud que se muestra cuando se ejecuta el mandato **runmqsc**. Consulte [“Establecimiento del indicador de mandatos MQSC”](#) en la página 15.

 Al ejecutar mandatos MQSC de forma interactiva en plataformas UNIX and Linux, la línea de mandatos **runmqsc** admite la recuperación de mandatos, la finalización de mandatos y las teclas de mandato Emacs. Consulte [“El mandato runmqsc en UNIX y Linux”](#) en la página 16

### Acerca de esta tarea

### Procedimiento

1. Para utilizar los mandatos MQSC de forma interactiva, abra una ventana de mandatos o un shell y especifique el mandato siguiente:

```
runmqsc QMgrName
```

Donde *NombreGestorColas* especifica el nombre del gestor de colas que desea que procese los mandatos MQSC. Puede dejar *NombreGestorColas* en blanco para procesar mandatos MQSC en el gestor de colas predeterminado.

2. Escriba en los mandatos MQSC, según sea necesario. Por ejemplo, para crear una cola local llamada ORANGE . LOCAL . QUEUE, especifique el mandato siguiente:

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

En los mandatos que tienen demasiados parámetros y no caben en una sola línea, utilice los caracteres de continuación para indicar que un mandato continúa en la línea siguiente:

- Un signo menos (-) indica que el mandato debe continuar desde el principio de la siguiente línea.
- Un signo más (+) indica que el mandato debe continuar desde el primer carácter de la línea siguiente que no sea un blanco.

La entrada de mandatos termina con el último carácter que no sea un carácter de continuación de una línea que no esté en blanco. También puede terminar explícitamente la entrada de un mandato con un punto y coma (;).

3. Deje de trabajar con mandatos MQSC especificando el mandato siguiente:

```
end
```

Alternativamente, puede utilizar el carácter EOF correspondiente al sistema operativo.

## Resultados

Cuando emite mandatos MQSC, el gestor de colas devuelve mensajes de operador que confirman sus acciones o que le indican los errores que ha cometido. Por ejemplo, el mensaje siguiente confirma que se ha creado una cola:

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

El mensaje siguiente indica que ha realizado un error de sintaxis:

```
AMQ8405: Syntax error detected at or near end of command segment below:-  
AMQ8426: Valid MQSC commands are:
```

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

Estos mensajes se envían al dispositivo de salida estándar. Si no ha especificado el mandato correctamente, consulte la información de referencia para que el mandato encuentre la sintaxis correcta. Consulte [Mandatos MQSC](#).

### Tareas relacionadas

“Ejecución de mandatos MQSC desde archivos de texto” en la [página 17](#)

La ejecución de mandatos MQSC de forma interactiva es adecuada para pruebas rápidas, pero si tiene mandatos muy largos o utiliza una secuencia determinada de mandatos repetidamente, puede redirigir stdín desde un archivo de texto. También puede redirigir la salida a un archivo.

### Referencia relacionada

[runmqsc](#)

## Establecimiento del indicador de mandatos MQSC

Puede establecer el indicador de mandatos MQSC en una solicitud de su elección utilizando la variable de entorno **MQPROMPT**.

### Procedimiento

- Establezca la variable de entorno **MQPROMPT** en la solicitud de su elección.

La solicitud se inserta tanto cuando la consola de **runmqsc** se ejecuta interactivamente y cuando la entrada se redirige a **runmqsc** desde un archivo o desde el dispositivo de entrada estándar (stdín).

Puede incluir texto sin formato en el indicador de mandatos y también puede insertar variables de entorno utilizando la notación +VARNAME+ del mismo modo que las definiciones de objeto de servicio de IBM MQ. Para obtener más información, consulte [“Definición de un objeto de servicio” en la página 195](#).

Por ejemplo, para establecer la solicitud en MQSC, especifique uno de los mandatos siguientes:

## Windows

```
set MQPROMPT="MQSC"
```

## Linux UNIX

```
export MQPROMPT="MQSC"
```

### Ejemplo

El ejemplo siguiente muestra el establecimiento de la variable **MQPROMPT** en un sistema AIX. El indicador se establece para mostrar un nombre de usuario, el nombre del gestor de colas y el nombre de host de IBM MQ. Estos valores se toman de otras variables de entorno que están establecidas en el sistema:

```
sh> export MQPROMPT="+USER+ @ +QMNAME+ @ +MQ_HOST_NAME+> "  
sh> runmqsc MY.QMGR  
5724-H72 (C) Copyright IBM Corp. 1994, 2024.  
Starting MQSC for queue manager MY.QMGR.  
myuser @ MYQMGR @ aix1> DISPLAY QMSTATUS
```

## Linux

## UNIX

### El mandato **runmqsc** en UNIX y Linux

La línea de mandatos **runmqsc** en UNIX y Linux da soporte a la recuperación de mandatos, la finalización del mandato y las teclas de mandato Emacs.

Están disponibles las funciones de editor de línea de mandatos siguientes:

- Recuperación de mandatos especificados anteriormente utilizando la tecla de flecha arriba y la tecla de flecha abajo
- Capacidad de completar automáticamente la siguiente palabra clave de un mandato utilizando la tecla de tabulación y la barra espaciadora
- Teclas de mandato de Emacs o funciones de teclas de mandato similares

Para utilizar estas funciones, la biblioteca de curses debe estar instalada. Si la biblioteca de curses no está instalada en el sistema, **runmqsc** no tendrá las funciones de editor de línea de mandatos y se mostrará un mensaje cuando se inicie la línea de mandatos **runmqsc**. El nombre de la biblioteca de curses a instalar dependerá de la plataforma UNIX:

- **AIX** En AIX, instale curses
- En todas las demás plataformas, y en Linux, instale ncurses

### Instalación de **ncurses** o **curses** en AIX

**Nota:** El ejemplo siguiente utiliza instrucciones para Linux

Ejecute el mandato siguiente para buscar los paquetes de ncurses existentes:

```
rpm -qa | grep -i ncurses
```

Los paquetes de ncurses necesarios son los siguientes:

```
ncurses-term-6.1-7.20180224.e18.noarch  
ncurses-6.1-7.20180224.e18.x86_64  
ncurses-base-6.1-7.20180224.e18.noarch  
ncurses-c++-libs-6.1-7.20180224.e18.x86_64  
ncurses-libs-6.1-7.20180224.e18.x86_64  
ncurses-compat-libs-6.1-7.20180224.e18.x86_64  
ncurses-devel-6.1-7.20180224.e18.x86_64
```



Puede instalar todos los paquetes de ncurses necesarios listados en el texto anterior ejecutando el mandato siguiente:

```
yum install ncurses*
```

## Personalización de los enlaces de teclas de Emacs

Puede personalizar las teclas que están enlazadas a los mandatos. Por ejemplo, puede enlazar las teclas a enlaces vi en lugar de los enlaces de teclas de Emacs personalizadas.

Las claves se personalizan editando el archivo `.editrc` que se almacena en el directorio inicial. Para obtener más información, consulte [editoric](#) en las páginas FreeBSD man.

## Inhabilitación de la recuperación de mandatos, la posibilidad de completar mandatos y las teclas de mandatos de Emacs

Puede inhabilitar la recuperación de mandatos, la posibilidad de completar mandatos y las teclas de mandatos de Emacs estableciendo una variable de entorno. Establezca la variable de entorno **MQ\_OVERRIDE\_LIBEDIT\_LOAD** en TRUE.

Esta variable de entorno se puede utilizar como método alternativo cuando **runmqsc** muestra el siguiente mensaje informativo:

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

## Ejecución de mandatos MQSC desde archivos de texto




La ejecución de mandatos MQSC de forma interactiva es adecuada para pruebas rápidas, pero si tiene mandatos muy largos o utiliza una secuencia determinada de mandatos repetidamente, puede redirigir `stdin` desde un archivo de texto. También puede redirigir la salida a un archivo.

### Acerca de esta tarea

La entrada para el mandato **runmqsc** se toma del *dispositivo de entrada estándar*, que también se conoce como `stdin`. `stdin` es el dispositivo desde el que se toma la entrada al sistema. Normalmente es el teclado, pero puede especificar que la entrada puede proceder de un puerto serie o de un archivo de disco, por ejemplo.

La salida para el mandato **runmqsc** es la salida del *dispositivo de salida estándar*, que se también se conoce como `stdout`. `stdout` es el dispositivo al que se envía la salida del sistema. Normalmente es una pantalla, pero puede redirigir la salida a un puerto serie o a un archivo.

Puede interesarle colocar los mandatos MQSC en un script cuando utilice:

-  Los archivos de inicialización CSQINP1, CSQINP2 y CSQINPX o el programa de utilidad de proceso por lotes CSQUTIL en z/OS.
-  El mandato **STRMQM** en IBM i.
-  El mandato **runmqsc** en UNIX, Linux, and Windows.

Puede establecer el indicador de mandatos MQSC en una solicitud de su elección utilizando la variable de entorno **MQPROMPT**. Para obtener más información, consulte [“Establecimiento del indicador de mandatos MQSC”](#) en la [página 15](#).

## Procedimiento

1. Cree un archivo de texto que contenga los mandatos MQSC que desea ejecutar.

- Por motivos de portabilidad entre los entornos IBM MQ, limite la longitud de línea en los archivos de mandatos MQSC a 72 caracteres.
- Cada mandato debe empezar en una línea nueva.
- Una línea que tenga un asterisco (\*) en la primera posición se ignora. Esto se puede usar para insertar comentarios en el archivo.
- Las líneas en blanco se ignoran.
- Un signo más (+) indica que el mandato continúa desde el primer carácter que no sea un espacio en blanco en la línea siguiente. Si utiliza + para continuar un mandato, recuerde que debe dejar al menos un espacio en blanco antes del siguiente parámetro (excepto en z/OS donde no es necesario). Los comentarios o las líneas en blanco se descartan cuando el mandato se vuelve a ensamblar en una sola serie.
- Un signo menos (-) indica que el mandato va a continuar desde el principio de la línea siguiente. Los comentarios o las líneas en blanco se descartan cuando el mandato se vuelve a ensamblar en una sola serie.
- Los mandatos MQSC que están contenidos en un mandato Escape PCF (formato de mandato programable) no pueden continuar con el signo más o el signo menos. Todo el mandato debe estar contenido en un solo mandato Escape. (Para obtener información acerca de los mandatos PCF, consulte [Introducción a los formatos de mandato programable](#)).
- En Multiplatforms, y en z/OS para los mandatos emitidos desde el programa de utilidad de proceso por lotes CSQUTIL, puede utilizar un carácter de punto y coma (;) para terminar un mandato, incluso si ha especificado un signo más (+) al final de la línea anterior.
- Una línea no debe terminar en un carácter de control del teclado (por ejemplo, un tabulador).
- Si ejecuta el mandato **runmqsc** en modalidad de cliente redirigiendo stdin desde un archivo de texto, IBM MQ espera que la primera línea del archivo de entrada sea una contraseña.
- **Windows** En Windows, si determinados caracteres especiales como el signo de la libra (£) y el NOT lógico (¬) se utilizan en un script de mandato (por ejemplo, como parte de una descripción de objeto), se visualizan de forma distinta en la salida de un mandato como, por ejemplo, **DISPLAY QLOCAL**.
- Si desea más información sobre la sintaxis del mandato MQSC, consulte [Mandatos MQSC](#).
- Puede utilizar los archivos de mandatos MQSC de ejemplo para ayudarle a crear el archivo de texto.

#### **amqscos0.tst**

Definiciones de objetos utilizadas por programas de ejemplo.

#### **amqscic0.tst**

Definiciones de colas para transacciones CICS.

**Windows** En Windows, estos archivos se encuentran en el directorio `MQ_INSTALLATION_PATH\tools\mqsc\samples`. `MQ_INSTALLATION_PATH` representa el directorio de alto nivel en el que está instalado IBM MQ.

**Linux** **UNIX** En UNIX and Linux, estos archivos se encuentran en el directorio `MQ_INSTALLATION_PATH/samp`. `MQ_INSTALLATION_PATH` representa el directorio de alto nivel en el que está instalado IBM MQ.

2. Verifique que la sintaxis del mandato es correcta en un gestor de colas local sin ejecutar los mandatos. Utilice el distintivo **-v** en el mandato **runmqsc**:

Realice una de las siguientes acciones:

- **V 9.1.4** **CD** Utilice la opción **-f** para indicar el nombre de archivo de texto:

```
runmqsc -f myprog.in -v QmgrName
```

o

- ▶ **LTS** Antes de IBM MQ 9.1.4 , utilice el operador < para dirigir los mandatos MQSC desde el archivo de texto myprog.in al mandato:

```
runmqsc -v QmgrName < myprog.in
```

El informe que se devuelve es similar al que se muestra en la [Figura 2](#) en la [página 20](#).

No puede especificar un gestor de colas remoto al verificar mandatos. Es decir, no puede especificar el distintivo **-w**.

### 3. Cuando se utiliza el mandato **runmqsc**:

Realice una de las siguientes acciones:

- ▶ **V 9.1.4** ▶ **CD** Utilice la opción **-f** en el mandato runmqsc para identificar el archivo de entrada, por ejemplo, myprog.in

```
runmqsc -f myprog.in -v QmgrName
```

o

- ▶ **LTS** Anterior a IBM MQ 9.1.4 utilice

- El operador < para direccionar la entrada desde un archivo de texto. Por ejemplo, el mandato siguiente ejecuta una secuencia de mandatos contenidos en el archivo de texto myprog.in:

```
runmqsc QMgrName < myprog.in
```

, o

- El operador > para direccionar la salida a un archivo de texto. Por ejemplo, el mandato siguiente ejecuta una secuencia de mandatos contenidos en el archivo de texto myprog.in y lo envía a un archivo denominado results.out:

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

[Figura 1](#) en la [página 19](#) muestra un extracto del archivo de mandatos MQSC myprog.in y [Figura 2](#) en la [página 20](#) muestra el extracto correspondiente de la salida en myprog.out.

### Ejemplo

Los mandatos MQSC se escriben en formato que puede leer el usuario, es decir, en texto ASCII. El ejemplo siguiente es un extracto de un archivo de mandatos MQSC que muestra un mandato MQSC, **DEFINE QLOCAL**.

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

*Figura 1. Extracto de un archivo de mandatos MQSC*

Cuando el mandato **runmqsc** se completa, se devuelve un informe. El ejemplo siguiente es un extracto de un informe:

```

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

```

Figura 2. Extracto de un archivo de informe de mandatos MQSC

### Tareas relacionadas

“Establecimiento del indicador de mandatos MQSC” en la página 15

Puede establecer el indicador de mandatos MQSC en una solicitud de su elección utilizando la variable de entorno **MQPROMPT**.

“Ejecución interactiva de mandatos MQSC” en la página 14

Puede utilizar los mandatos MQSC de forma interactiva, mediante una ventana de mandatos o shell.

### Referencia relacionada

[runmqsc](#)

z/OS

## Orígenes desde los que puede emitir mandatos MQSC en z/OS

Los mandatos MQSC se pueden emitir desde varios orígenes, en función del mandato.

Los mandatos pueden emitirse desde las siguientes fuentes:

- La consola de z/OS o equivalente
- Los conjuntos de datos de entrada de inicialización CSQINP1, CSQINP2, CSQINPT y CSQINPX
- El programa de utilidad de proceso por lotes CSQUTIL
- Aplicaciones adecuadamente autorizadas, enviando mandatos como mensajes a la cola SYSTEM.COMMAND.INPUT

z/OS

Para obtener más detalles, consulte [Emisión de mandatos](#),

No obstante, no todos los mandatos pueden emitirse desde todas estas fuentes. Los mandatos se pueden clasificar según desde dónde se pueden emitirse:

**1**

CSQINP1

**2**

CSQINP2

**C**

La consola de z/OS

**R**

El servidor de mandatos y la cola de mandato, por medio de CSQUTIL, CSQINPT, CSQINPX, o aplicaciones

Dentro de las descripciones de mandatos en [Mandatos MQSC](#), estos orígenes se identifican mediante el uso de los caracteres 1, 2, C y R en cada descripción de mandato.

## Configuración automática de un script de mandato de script de WebSphere MQ en el inicio

A partir de IBM MQ 9.1.4, puede configurar el gestor de colas para que aplique automáticamente el contenido de un script de mandato de script de WebSphere MQ, o un conjunto de scripts mandato de script de WebSphere MQ, en cada inicio del gestor de colas.

Puede utilizar esta funcionalidad para tener una configuración que se pueda modificar y se reproduzca automáticamente en el siguiente reinicio del gestor de colas. Como ejemplo, si el script o los scripts se encuentran en un disco montado, es posible tener una configuración centralizada en la que se aplica la última versión a cada gestor de colas a medida que se inician.

Un escenario en particular en el que puede ser de utilidad, es para asegurarse de que un clúster uniforme contiene las mismas definiciones en todos los gestores de colas del clúster, teniendo un conjunto único de configuración que todos aplican. Para obtener un ejemplo de esto, consulte [Creación de un clúster uniforme desde IBM MQ 9.1.4](#).

### Antes de empezar

Puede utilizar:

1. Un único script y crear un archivo de texto utilizando los mandatos MQSC.
2. Un conjunto de scripts MQSC:
  - Para identificar un directorio en el que existirán las configuraciones
  - En dicho directorio, cree archivos, cada uno con la extensión `.mqsc`, por ejemplo `queues.mqsc`.

Dado que este script se vuelve a aplicar en cada inicio del gestor de colas, es importante que se puedan reproducir los mandatos. Por ejemplo, un mandato **DEFINE** debe incluir la serie **REPLACE**, de lo contrario, el mandato aparecerá como una anomalía en el segundo inicio del gestor de colas, porque el objeto ya existe.

Tenga en cuenta que en un script de mandato de script de WebSphere MQ, cualquier línea con un `*` antepuesto se trata como un comentario.

### Habilitación de la configuración automática de scripts de mandato de script de WebSphere MQ

Puede configurar un nuevo gestor de colas utilizando el distintivo **-ic** para el mandato `crtmqm` y apuntando a un archivo o directorio específico. El valor suministrado se almacena en el archivo `qm.ini` bajo la stanza `AutoConfig`, como el atributo **MQSCConfig**.

Para configurar un gestor de colas existente para habilitar la configuración de mandato de script de WebSphere MQ automática, añada el atributo de stanza `AutoConfig` **MQSCConfig**, que apunte a un archivo o directorio válido. Por ejemplo:

```
AutoConfig:
MQSCConfig=C:\mq_configuration\uniclus.mqsc
```

### ¿Cómo funciona el trabajo de configuración automática?

Durante el inicio del gestor de colas, la configuración identificada por el atributo de stanza `AutoConfig` **MQSCConfig** se pasa a través de la validación `runmqsc`, para garantizar que la sintaxis es válida y, a continuación, se almacena en el árbol de datos del gestor de colas en el directorio `autocfg` como un único archivo `cached.mqsc`.

Cuando se procesan varios archivos de un directorio, se procesan en orden alfabético y, si contiene un mandato `end` o `quit` de MQSC, el resto del contenido de dicho archivo se omite.

Durante el primer inicio del gestor de colas, la incapacidad para leer el archivo o el directorio, o un archivo con sintaxis de mandato de script de WebSphere MQ que no es válida, impide que se inicie el gestor de colas, con un mensaje de error adecuado en el registro cronológico de errores de la consola y del gestor de colas.

En los reinicios posteriores, si el archivo o el directorio al que se apunta es ilegible o contiene una sintaxis de mandato de script de WebSphere MQ no válida, se utiliza el archivo almacenado en memoria caché anteriormente y se destaca esto en un mensaje grabado en el registro cronológico de errores del gestor de colas.

En el punto en que el gestor de colas está habilitado para que las aplicaciones se conecten, el contenido de `cached.mqsc` se aplica al gestor de colas. El registro **runmqsc** de la configuración que se está aplicando se almacena en el directorio de errores del gestor de colas, como un archivo llamado `autocfgmqsc.LOG`.

Además, cualquier mandato de script de WebSphere MQ que no se complete correctamente, se registra en el registro cronológico de errores del gestor de colas, e identifica porqué falla el mandato.

## Resolución de problemas con mandatos MQSC

Si no consigue que se ejecuten los mandatos MQSC, utilice la información de este tema para ver si es debido a alguno de estos problemas comunes. La causa del problema no siempre es evidente cuando se lee el error que un genera un mandato.

### Procedimiento

- Si está redirigiendo la entrada de un archivo, asegúrese de que utiliza el operador de redirección `<`. Si omite este operador, el gestor de colas interpretará el nombre del archivo como si se tratase de un nombre de gestor de colas y emitirá el siguiente mensaje de error:

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

- Si está redirigiendo la salida a un archivo, asegúrese de que utiliza el operador de redirección `>`. De forma predeterminada, el archivo se coloca en el directorio de trabajo actual al invocar **runmqsc**. Especifique un nombre de archivo completamente calificado para enviar la salida a un archivo y directorio específicos.
- Compruebe que ha creado el gestor de colas que va a ejecutar los mandatos, y que el gestor de colas se está ejecutando, utilizando el mandato siguiente para mostrar todos los gestores de colas:

```
dspmq
```

Si el gestor de colas no se está ejecutando, inícielo. Para obtener más información, consulte [“Inicio de un gestor de colas”](#) en la página 142.

- Asegúrese de que ha especificado un gestor de colas en el mandato **runmqsc** o de que tiene un gestor de colas predeterminado definido.

Si no ha definido un gestor de colas predefinido, u obtiene este error:

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

- Compruebe que no está intentando especificar un mandato MQSC como un parámetro del mandato **runmqsc**. No puede especificar un mandato MQSC como parámetro del mandato **runmqsc**. Por ejemplo, esto no es válido:

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

- Compruebe que ha emitido el mandato **runmqsc**. No puede entrar mandatos MQSC antes de emitir el mandato **runmqsc**.

- Compruebe que no está intentando ejecutar un mandato de control desde **runmqsc**. No puede ejecutar mandatos de control desde **runmqsc**. Por ejemplo, no puede emitir el mandato **strmqm** para iniciar un gestor de colas mientras está ejecutando mandatos MQSC interactivamente. Si lo hace, recibirá mensajes de error similares al mensaje siguiente:

```
runmqsc
.
.
Starting MQSC for queue manager jupiter.queue.manager.
1 : strmqm saturn.queue.manager
AMQ8405: Syntax error detected at or near end of cmd segment below:-s
AMQ8426: Valid MQSC commands are:
ALTER
CLEAR
DEFINE
DELETE
DISPLAY
END
PING
REFRESH
RESET
RESOLVE
RESUME
START
STOP
SUSPEND
2 : end
```

## Valores y caracteres genéricos con significados especiales

La información siguiente describe los valores genéricos y los caracteres que tienen un significado especial al crear mandatos de MQSC.

Siempre que un parámetro pueda tener un valor genérico, se especifica con un asterisco (\*) final, por ejemplo, ABC\*. Un valor genérico significa todos los valores que empiezan por; por lo tanto, ABC\* significa todos los valores que empiezan por ABC.

Si en el valor se utilizan caracteres que requieren comillas, se debe colocar el asterisco dentro de las comillas, por ejemplo, 'abc\*'. El asterisco debe ser el último, o el único, carácter del valor.

No se permite usar signos de cerrar interrogación (?) y de dos puntos (:) en los valores genéricos.


Carácter	Descripción
	Los espacios en blanco se usan como separadores. Varios espacios en blanco son equivalentes a un solo espacio en blanco, excepto en series que van entre póstrfos (!). Los blancos finales de estos atributos de serie basados en tipos MQCHARV se tratan como significativos.
,	Se utilizan comas como separadores. Varias comas equivalen a una sola coma, excepto en las series encerradas entre apóstrofos (!).
'	Un apóstrofo indica el principio o el final de una serie. IBM MQ deja todos los caracteres encerrados entre comillas exactamente como se han introducido. Los apóstrofos incluidos no se incluyen al calcular la longitud de la serie.
"	IBM MQ trata las comillas simples que aparecen dentro de una serie como un carácter al calcular la longitud de la serie y la serie no ha terminado.
=	 En z/OS, un signo de igual indica el inicio del valor de un parámetro que termina con una coma o un espacio en blanco.
(	Un signo de abrir paréntesis indica el principio del valor o la lista de valores de un parámetro.

Tabla 1. Descripciones de caracteres que tienen significados especiales (continuación)

Carácter	Descripción
)	Un signo de cerrar paréntesis indica el final del valor o la lista de valores de un parámetro.
:	Los dos puntos indican un rango inclusivo. Por ejemplo (1:5) significa (1,2,3,4,5). Esta notación solo se puede utilizar en los mandatos <b>TRACE</b> .
*	Un asterisco significa todo. Por ejemplo, <b>DISPLAY TRACE (*)</b> significa mostrar todos los rastreos y <b>DISPLAY QUEUE (PAY*)</b> significa mostrar todas las colas con nombres que empiezan por PAY.

Cuando necesite utilizar cualquiera de estos caracteres especiales en un campo (por ejemplo, como parte de una descripción), deberá delimitar la serie completa entre comillas simples.

## Automatización de la administración de IBM MQ utilizando mandatos PCF

Puede decidir que podría ser beneficioso para su instalación automatizar algunas tareas de administración y supervisión. Puede automatizar tareas de administración para los gestores de colas locales y remotos utilizando mandatos de formato de mandato programable (PCF). En esta sección se presupone que tiene experiencia en la administración de objetos de IBM MQ.

### mandatos PCF

Los mandatos PCF (formato de mandato programable) de IBM MQ se pueden utilizar para programar tareas de administración en un programa de administración. De este modo, desde un programa puede manipular objetos de gestores de colas (colas, definiciones de procesos, listas de nombres, canales, canales de conexión de clientes, escuchas, servicios y objetos de información de autenticación), e incluso manipular los mismos gestores de colas.

Los mandatos PCF abarcan el mismo tipo de funciones que las proporcionadas por los mandatos MQSC. Puede escribir un programa que emita mandatos PCF a cualquier gestor de colas de la red desde un solo nodo. De este modo, puede centralizar y automatizar las tareas de administración.

Cada mandato PCF es una estructura de datos que se incluye en la parte de datos de aplicación de un mensaje de IBM MQ. Cada mandato se envía al gestor de colas de destino utilizando la función MQPUT de MQI del mismo modo que cualquier otro mensaje. Si el servidor de mandatos se está ejecutando en el gestor de colas que recibe el mensaje, el servidor de mandatos lo interpreta como un mensaje de mandato y ejecuta el mandato. Para obtener las respuestas, la aplicación emite una llamada MQGET y los datos de respuesta se devuelven en otra estructura de datos. La aplicación puede entonces procesar la respuesta y actuar en conformidad.

**Nota:** A diferencia de los mandatos MQSC, los mandatos PCF y sus respuestas no están en un formato de texto legible por el usuario.

En resumen, estas son algunas de las cosas necesarias para crear un mensaje de mandato PCF:

#### Descriptor de mensaje

Es un descriptor de mensaje estándar de IBM MQ, en el que:

- El tipo de mensaje (*MsqType*) es MQMT\_REQUEST.
- El formato del mensaje (*Format*) es MQFMT\_ADMIN.

#### Datos de aplicación

Contienen el mensaje PCF, incluida la cabecera PCF, en el que:

- El tipo de mensaje PCF (*Type*) especifica MQCFT\_COMMAND.
- El identificador de mandato especifica el mandato, por ejemplo, *Change Queue* (MQCMD\_CHANGE\_Q).



Para obtener una descripción completa de las estructuras de datos PCF y cómo implementarlas, consulte [“Introducción a los Formatos de mandato programable de IBM MQ” en la página 25.](#)

## Atributos de objetos PCF

Los atributos de objetos en PCF no están limitados a ocho caracteres, como sí lo están para los mandatos MQSC. Se muestran en esta guía en cursiva. Por ejemplo, el equivalente PCF de RQMNAME es *RemoteQMGrName*.

## PCF de escape

Los PCF de escape son mandatos PCF que contienen mandatos MQSC dentro del texto del mensaje. Los PCF se pueden utilizar para enviar mandatos a un gestor de colas remoto. Para obtener más información sobre los PCF de escape, consulte [Escape](#).

## Introducción a los Formatos de mandato programable de IBM MQ

Los formatos de mandato programable (PCF) definen mensajes de mandato y de respuesta que se pueden intercambiar entre un programa y cualquier gestor de colas (que admita PCF) en una red. Los PCF simplifican la administración del gestor de colas y otras tareas de administración de red. Pueden utilizarse para resolver el problema de la compleja administración de redes distribuidas, especialmente cuando las redes crecen en tamaño y complejidad.

Los formatos de mandatos programables están soportados por:

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

## Problema que resuelven los mandatos PCF

La administración de redes distribuidas puede ser compleja. Los problemas de administración continúan creciendo a medida que las redes presentan mayor tamaño y complejidad.

Como ejemplos de administración específicos de mensajes y colas, cabe citar:

- Gestión de recursos.

Por ejemplo, la creación y supresión de colas.

- Supervisión del rendimiento.

Por ejemplo, una mayor profundidad de cola o un mayor índice de mensajes.

- Control.

Por ejemplo, ajustar los parámetros de colas tales como la profundidad máxima de cola, la longitud máxima de los mensajes, y la habilitación e inhabilitación de colas.

- Direccionamiento de mensajes.

Definición de rutas alternativas a través de una red.

Los mandatos PCF de IBM MQ pueden utilizarse para simplificar la administración del gestor de colas y otras tareas de administración de la red. Los mandatos PCF permiten utilizar una única aplicación para realizar la administración de red desde un único gestor de colas en la red.

## ¿Qué son los PCF?

Los PCF definen mensajes de mandato y de respuesta que se pueden intercambiar entre un programa y cualquier gestor de colas (que admita PCF) en una red. Puede utilizar los mandatos PCF en el programa de aplicación de gestión de sistemas para la administración de objetos de IBM MQ: los objetos de información de autenticación, los canales, los escuchas de canales, las listas de nombres, las definiciones de proceso, los gestores de colas, las colas, los servicios y las clases de almacenamiento. La aplicación puede operar desde un único punto de la red para comunicar información de mandato y de respuesta a cualquier gestor de colas, local o remoto, utilizando el gestor de colas local.


Cada gestor de colas tiene una cola de administración con un nombre de cola estándar y la aplicación puede enviar mensajes de mandato PCF a dicha cola. Cada gestor de colas tiene también un servidor de mandatos para prestar servicio a los mensajes de mandato desde la cola de administración. Los mensajes de mandatos PCF pueden, por consiguiente, ser procesados por cualquier gestor de colas de la red y los datos de respuesta pueden devolverse a la aplicación mediante la cola de respuesta especificada. Los mensajes de mandatos PCF y de respuesta se envían y reciben utilizando la Interfaz de colas de mensajes (MQI) habitual.

Para obtener una lista de los mandatos PCF disponibles, incluidos sus parámetros, consulte [Definiciones de los formatos de mandato programable](#).

## Utilización de los formatos de mandato programable de IBM MQ

Puede utilizar los PCF en un programa de gestión de sistemas para la administración remota de IBM MQ.

Esta sección incluye:


- [“Mensajes de mandato PCF” en la página 26](#)
- [“Respuestas PCF en IBM MQ” en la página 29](#)
-  [“Respuestas ampliadas” en la página 31](#)
- [Reglas de denominación de objetos de IBM MQ](#)
- [“Comprobación de autorización para mandatos PCF en IBM MQ” en la página 33](#)


### Mensajes de mandato PCF

Los mensajes de mandato PCF constan de una cabecera PCF, parámetros identificados en dicha cabecera y los datos de mensaje definidos por el usuario. Los mensajes se emiten mediante llamadas de la interfaz de cola de mensajes.

Cada mandato y sus parámetros se envían como un mensaje de mandato independiente que contiene una cabecera PCF seguida de varias estructuras de parámetros; para obtener más información sobre la cabecera PCF, consulte [MQCFH - Cabecera PCF](#) y para ver un ejemplo de una estructura de parámetros, consulte [MQCFST - Parámetro de serie PCF](#). La cabecera PCF identifica el mandato y las estructuras de parámetro que siguen en el mismo mensaje. Cada estructura de mandato proporciona un parámetro al mandato.

Las respuestas a los mandatos, generadas por el servidor de mandatos, tienen una estructura similar. Hay una cabecera PCF, seguida de varias estructuras de parámetros. Las respuestas pueden constar de más de un mensaje, pero los mandatos siempre constan de un único mensaje.

 En Multiplatforms, la cola a la que se envían los mandatos PCF siempre se denomina `SYSTEM.ADMIN.COMMAND.QUEUE`.

 En z/OS, los mandatos se envían a `SYSTEM.COMMAND.INPUT`, aunque `SYSTEM.ADMIN.COMMAND.QUEUE` puede ser un alias para él. El servidor de mandatos que da servicio a esta cola envía las respuestas a la cola definida por los campos *ReplyToQ* y *ReplyToQMgr* en el descriptor de mensaje del mensaje de mandato.

## Cómo emitir mensajes de mandatos PCF

Utilice las llamadas normales de la interfaz de cola de mensajes (MQI), MQPUT, MQGET, etc., para colocar y recuperar mensajes de respuestas y de mandatos PCF en y desde sus colas.

### Nota:

Asegúrese de que el servidor de mandatos se ejecuta en el gestor de colas de destino para que el mandato PCF se procese en dicho gestor de colas.

Para obtener una lista de los archivos de cabecera suministrados, consulte [Archivos COPY, de cabecera, de inclusión y de módulo de IBM MQ](#).

## Descriptor de mensaje para un mandato PCF

El descriptor de mensaje de IBM MQ está completamente documentado en [MQMD - Descriptor de mensaje](#).

Un mensaje de mandato PCF contiene los siguientes campos en el descriptor de mensaje:

### Informe

Cualquier valor válido, según sea necesario.

### MsgType

Este campo debe ser MQMT\_REQUEST para indicar un mensaje que requiere una respuesta.

### Caducidad

Cualquier valor válido, según sea necesario.

### Comentarios

Se establece en MQFB\_NONE.

### Codificación

Si está enviando a uno de los sistemas siguientes, establezca este campo a la codificación utilizada en los datos del mensaje; la conversión se lleva a cabo si es necesario:

-  IBM i
-  Linux
-  UNIX
-  Windows

### CodedCharSetId

Si está enviando a uno de los sistemas siguientes, establezca este campo al identificador de juego de caracteres codificado que se usa en los datos del mensaje; la conversión se lleva a cabo si es necesario:

-  IBM i
-  Linux
-  UNIX
-  Windows

### Formato

Se establece en MQFMT\_ADMIN.

### Priority

Cualquier valor válido, según sea necesario.

### Persistence

Cualquier valor válido, según sea necesario.

**MsgId**

La aplicación de envío puede especificar cualquier valor, o puede especificarse MQMI\_NONE para solicitar al gestor de colas que genere un identificador de mensaje exclusivo.

**CorrelId**

La aplicación emisora puede especificar cualquier valor, o puede especificarse MQCI\_NONE para indicar que no hay identificador de correlación.

**ReplyToQ**

Nombre de la cola que debe recibir la respuesta.

**ReplyToQMGr**

Nombre del gestor de colas para la respuesta (o en blanco).

**Campos de contexto de mensaje**

Estos campos pueden establecerse en cualquier valor válido, según sea necesario. Normalmente, la opción MQPMO\_DEFAULT\_CONTEXT se utiliza para establecer los campos de contexto de mensaje en los valores predeterminados.

Si utiliza una estructura MQMD versión 2, debe establecer los siguientes campos adicionales:

**GroupId**

Se establece en MQGI\_NONE.

**MsgSeqNumber**

Se establece en 1.

**Desplazamiento**

Se establece en 0.

**MsgFlags**

Se establece en MQMF\_NONE.

**OriginalLength**

Se establece en MQOL\_UNDEFINED.

**Envío de datos de usuario**

Las estructuras PCF también pueden utilizarse para enviar datos de mensaje definidos por el usuario. En este caso, el campo *Format* del descriptor de mensaje debe establecerse en MQFMT\_PCF.

***Envío y recepción de mensajes PCF en una cola especificada*****Envío de mensajes PCF a una cola especificada**

Para enviar un mensaje a una cola especificada, la llamada mqPutBag convierte el contenido del paquete especificado en un mensaje PCF y envía el mensaje a la cola especificada. El contenido del paquete permanece intacto después de la llamada.

Como entrada para esta llamada, debe proporcionar:

- Un manejador de conexión MQI.
- Un manejador de objeto para la cola en la que va a colocarse el mensaje.
- Un descriptor de mensaje. Para obtener más información sobre el descriptor de mensaje, consulte [MQMD - Descriptor de mensaje](#).
- Las opciones de transferencia de mensajes utilizando la estructura MQPMO. Para obtener más información acerca de la estructura MQPMO, consulte [MQPMO – Opciones de transferir mensaje](#).
- El manejador del paquete que se convertirá en un mensaje.

**Nota:** Si el paquete contiene un mensaje de administración y se utilizó la llamada mqAddInquiry para insertar valores en el paquete, el valor del elemento de datos MQIASY\_COMMAND debe ser un mandato INQUIRE reconocido por la MQAI.

Para obtener una descripción completa de la llamada mqPutBag, consulte [mqPutBag](#).

## Recepción de los mensajes PCF de una cola especificada

Para recibir un mensaje de una cola especificada, la llamada `mqGetBag` obtiene un mensaje PCF de una cola especificada y convierte los datos del mensaje en un paquete de datos.

Como entrada para esta llamada, debe proporcionar:

- Un manejador de conexión MQI.
- Un manejador de objeto de la cola de la que se debe leer el mensaje.
- Un descriptor de mensaje. Dentro de la estructura MQMD, el parámetro **Format** debe ser MQFMT\_ADMIN, MQFMT\_EVENT o MQFMT\_PCF.

**Nota:** Si el mensaje se recibe dentro de una unidad de trabajo (es decir, con la opción MQGMO\_SYNCPOINT) y el mensaje tiene un formato no soportado, la unidad de trabajo puede restituirse. El mensaje se reincorporará en la cola y podrá recuperarse utilizando la llamada MQGET en lugar de la llamada `mqGetBag`. Para obtener más información sobre el descriptor de mensaje, consulte [MQGMO - Opciones de obtención de mensajes](#).

- Las opciones de obtención de mensajes utilizando la estructura MQPMO. Para obtener más información acerca de la estructura MQGMO, consulte [MQMD - Descriptor de mensaje](#).
- El manejador del paquete para contener el mensaje convertido.

Para obtener una descripción completa de la llamada `mqGetBag`, consulte [mqGetBag](#).

### Respuestas PCF en IBM MQ

En respuesta a cada mandato, el servidor de mandatos genera uno o varios mensajes de respuesta. Un mensaje de respuesta tiene un formato parecido a un mensaje de mandato.

La cabecera PCF tiene el mismo valor de identificador de mandato que el mandato para el que es una respuesta (consulte MQCFH - Cabecera PCF para obtener más detalles). El identificador del mensaje y el identificador de correlación se establecen de acuerdo con las opciones de informe de la solicitud.

Si el tipo de cabecera PCF del mensaje de mandato es MQCFT\_COMMAND, sólo se generan respuestas estándar. Los mandatos de este tipo están soportados en todas las plataformas excepto z/OS. Las aplicaciones no dan soporte a PCF en z/OS; IBM MQ Windows Explorer es una de estas aplicaciones (no obstante, IBM WebSphere MQ 6.0 o posterior IBM MQ Explorer no da soporte a PCF en z/OS).

Si el tipo de cabecera PCF del mensaje de mandato es MQCFT\_COMMAND\_XR, se generan respuestas ampliadas o estándar. Los mandatos de este tipo están soportados en z/OS y en algunas otras plataformas. Los mandatos emitidos en z/OS sólo generan respuestas ampliadas. En otras plataformas, puede generarse cualquiera de los dos tipos de respuesta.

Si un mandato especifica un nombre de objeto genérico, se devuelve una respuesta independiente en su propio mensaje para cada objeto coincidente. Para la generación de respuestas, un mandato con un nombre genérico se trata como varios mandatos individuales (excepto para el campo de control MQCFC\_LAST o MQCFC\_NOT\_LAST). En caso contrario, un mensaje de mandato genera un mensaje de respuesta.

Determinadas respuestas PCF pueden devolver una estructura aun cuando no se solicite. Esta estructura se muestra en la definición de la respuesta ([Definiciones de los formatos de mandato programable](#)) como *siempre devuelta*. Esto se explica porque para estas respuestas es necesario indicar el nombre de los objetos en la respuesta para identificar el objeto al que se aplican los datos.

### Descriptor de mensaje para una respuesta

Un mensaje de respuesta tiene los siguientes campos en el descriptor de mensaje:

#### MsgType

Este campo es MQMT\_REPLY.

#### MsgId

Este campo se genera en el gestor de colas.

**CorrelId**

Este campo se genera de acuerdo con las opciones de informe del mensaje de mandato.

**Formato**

Este campo es MQFMT\_ADMIN.

**Codificación**

Se establece en MQENC\_NATIVE.

**CodedCharSetId**

Se establece en MQCCSI\_Q\_MGR.

**Persistence**

Igual que en el mensaje de mandato.

**Priority**

Igual que en el mensaje de mandato.

La respuesta se genera con MQPMO\_PASS\_IDENTITY\_CONTEXT.

*Respuestas estándar*

Los mensajes de mandato con el tipo de cabecera MQCFT\_COMMAND generan respuestas estándar. Los mandatos de este tipo están soportados en todas las plataformas excepto z/OS.

Hay tres tipos de respuesta estándar:

- respuesta OK
- Respuesta de error
- Respuesta de datos

**respuesta OK**

Esta respuesta consiste en un mensaje que empieza con una cabecera de formato de mandatos, con un campo *CompCode* con el valor MQCC\_OK o MQCC\_WARNING.

Para MQCC\_OK, el valor de *Reason* es MQRC\_NONE.

Para MQCC\_WARNING, *Reason* identifica la naturaleza de la advertencia. En este caso, la cabecera de formato de mandato puede ir seguida de una o más estructuras de parámetro de aviso apropiadas para este código de razón.

En cualquier caso, para un mandato de consulta es posible que sigan varias estructuras de parámetro, como se describe en las siguientes secciones.

**Respuesta de error**

Si el mandato tiene un error, se envía uno o varios mensajes de respuesta de error (es posible que se envíe más de uno incluso para un mandato que normalmente sólo tendría un mensaje de respuesta). Estos mensajes de error tienen el valor MQCFC\_LAST o MQCFC\_NOT\_LAST establecido, según corresponda.

Cada mensaje de este tipo empieza por una cabecera de formato de respuesta, con MQCC\_FAILED como valor de *CompCode* y un campo *Reason* que identifica el error en cuestión. Por lo general, cada mensaje describe un error distinto. Además, cada mensaje tiene ninguna o una estructura de parámetro de error (nunca más de una) después de la cabecera. Esta estructura de parámetros, si existe, es una estructura MQCFIN, con un campo *Parameter* que contiene uno de estos valores:

- MQIACF\_PARAMETER\_ID

El campo *Value* de la estructura es el identificador del parámetro que tenía el error (por ejemplo, MQCA\_Q\_NAME).

- MQIACF\_ERROR\_ID

Este valor se utiliza con MQRD\_UNEXPECTED\_ERROR como valor *Reason* (en la cabecera de formato de mandato). El campo *Value* de la estructura MQCFIN es el código de razón inesperado recibido por el servidor de mandatos.

- MQIACF\_SELECTOR

Este valor se produce si una estructura de lista (MQCFIL) enviada con el mandato contiene un selector duplicado o uno que no es válido. El campo *Reason* de la cabecera de formato de mandato identifica el error, y el campo *Value* de la estructura MQCFIN es el valor de parámetro de la estructura MQCFIL del mandato que presentaba el error.

- MQIACF\_ERROR\_OFFSET

Este valor se produce cuando se produce un error de comparación de datos en el mandato Sondear canal. El campo *Value* de la estructura es el desplazamiento del error de comparación de Ping Channel.

- MQIA\_CODED\_CHAR\_SET\_ID

Este valor se produce cuando el identificador del conjunto de caracteres codificado del descriptor de mensaje del mandato PCF entrante no coincide con el del gestor de colas de destino. El campo *Value* de la estructura es el identificador del conjunto de caracteres codificado del gestor de colas.

El último (o único) mensaje de respuesta de error es una respuesta de resumen, con MQCC\_FAILED como valor de *CompCode* y un campo *Reason* con el valor MQRCCF\_COMMAND\_FAILED. Este mensaje no tiene ninguna estructura de parámetro después de la cabecera.

## Respuesta de datos

Esta respuesta consta de una respuesta OK (como se ha descrito anteriormente) para un mandato de consulta. La respuesta OK va seguida por estructuras adicionales que contienen los datos solicitados como se describe en [Definiciones de los formatos de mandato programable](#).

Las aplicaciones no deben depender de que estas estructuras de parámetro adicionales se devuelvan en un orden determinado.

### Respuestas ampliadas

Los mandatos emitidos en z/OS generan respuestas ampliadas.

Hay tres tipos de respuesta ampliada:

- Respuesta de mensaje, con el tipo MQCFT\_XR\_MSG
- Respuesta de elemento, con el tipo MQCFT\_XR\_ITEM
- Respuesta de resumen, con el tipo MQCFT\_XR\_SUMMARY

Cada mandato puede generar uno o más conjuntos de las respuestas. Cada conjunto de respuestas comprende uno o varios mensajes, numerados secuencialmente a partir de 1 en el campo *MsgSeqNumber* de la cabecera PCF. El campo *Control* de la última (o única) respuesta de cada conjunto tiene el valor MQCFC\_LAST. Para todas las demás respuestas del conjunto, este valor es MQCFC\_NOT\_LAST.

Cualquier respuesta puede incluir una o varias estructuras MQCFBS opcionales en las que el campo *Parameter* se establece en MQBACF\_RESPONSE\_SET y el valor es un identificador del conjunto de respuestas. Los identificadores son exclusivos e identifican el conjunto de respuestas que contiene la respuestas. Para cada conjunto de respuestas existe una estructura MQCFBS que la identifica.

Las respuestas ampliadas tienen al menos dos estructuras de parámetro:

- Una estructura MQCFBS con el campo *Parameter* establecido en MQBACF\_RESPONSE\_ID. El valor de este campo es el identificador del conjunto de respuestas al que pertenece la respuesta. El identificador del primer conjunto es arbitrario. En los conjuntos posteriores, el identificador es un notificado con anterioridad en una estructura MQBACF\_RESPONSE\_SET.
- Una estructura MQCFST con el campo *Parameter* establecido MQCACF\_RESPONSE\_Q\_MGR\_NAME, cuyo valor es el nombre del gestor de colas desde el que proviene el conjunto de respuestas.

Muchas respuestas tienen estructuras de parámetro adicionales, que se describen en las secciones siguientes.

No se puede determinar de antemano cuántas respuestas hay en un conjunto, a menos que se obtengan respuestas hasta que se encuentre una con MQCFC\_LAST. Tampoco se puede determinar de antemano cuántos conjuntos de respuestas existen, porque cualquier conjunto puede incluir estructuras MQBACF\_RESPONSE\_SET para indicar que se generan conjuntos adicionales.

## Respuestas ampliadas para mandatos de consulta

Los mandatos de consulta suelen generar una respuesta de elemento (tipo MQCFT\_XR\_ITEM) para cada elemento encontrado que coincida con los criterios de búsqueda especificados. La respuesta de elemento tiene un campo *CompCode* en la cabecera con un valor MQCC\_OK, y un campo *Reason* con un valor MQRC\_NONE. También incluye otras estructuras de parámetro que describen el elemento y los atributos necesarios, como se describe en [Definiciones de los formatos de mandato programable](#).

Si un elemento contiene un error, el campo *CompCode* de la cabecera tiene el valor MQCC\_FAILED y el campo *Reason* identifica el error en cuestión. Se incluyen estructuras de parámetro adicionales para identificar el elemento.

Es posible que determinados mandatos de consulta devuelvan respuestas de mensaje generales (no específicas de un nombre) además de las respuestas de elemento. Estas respuestas son de carácter informativo o de error de tipo MQCFT\_XR\_MSG.

Si el mandato de consulta se ejecuta correctamente, es posible que obtenga una respuesta de resumen (tipo MQCFT\_XR\_SUMMARY), con MQCC\_OK como valor *CompCode*, y MQRC\_NONE como valor del campo *Reason*.

Si el mandato de consulta falla, es posible que se devuelvan respuestas de elemento, y, opcionalmente, es posible que obtenga una respuesta de resumen (tipo MQCFT\_XR\_SUMMARY), con MQCC\_FAILED como valor de *CompCode* y MQRCCF\_COMMAND\_FAILED como valor del campo *Reason*.

## Respuestas ampliadas para mandatos distintos del de consulta

Los mandatos de ejecución satisfactoria generan respuestas de mensajes en las que el campo *CompCode* de la cabecera tiene el valor MQCC\_OK, y el campo *Reason* tiene el valor MQRC\_NONE. Siempre hay como mínimo un mensaje; debe ser informativo (MQCFT\_XR\_MSG) o de resumen (MQCFT\_XR\_SUMMARY). Opcionalmente, puede haber mensajes informativos (tipo MQCFT\_XR\_MSG) adicionales. Cada mensaje informativo puede incluir varias estructuras de parámetros adicionales con información acerca del mandato; consulte la descripción de cada mandato para obtener información sobre las estructuras que pueden producirse.

Los mandatos que no se ejecutan correctamente generan respuestas de mensajes de error (tipo MQCFT\_XR\_MSG), en las que el campo *CompCode* de la cabecera tiene el valor MQCC\_FAILED y el campo *Reason* identifica el error en cuestión. Cada mensaje puede incluir varias estructuras de parámetros adicionales con información acerca del error; consulte la descripción de cada mandato para ver qué estructuras pueden producirse. Es posible que se generen respuestas de mensajes informativos. Opcionalmente, es posible que produzca una respuesta de resumen (MQCFT\_XR\_SUMMARY), con MQCC\_FAILED como valor de *CompCode* y MQRCCF\_COMMAND\_FAILED como valor del campo *Reason*.

## Respuestas ampliadas para mandatos utilizando CommandScope

Si un mandato utiliza el parámetro **CommandScope**, o hace que se genere un mandato que utilice el parámetro **CommandScope**, existe un conjunto de respuestas inicial del gestor de colas del que se ha recibido el mandato. A continuación, se genera uno o varios conjuntos de respuestas para cada gestor de colas al que se dirige el mandato (como si se emitieran varios mandatos individuales). Por último, hay un conjunto de respuestas del gestor de colas receptor que incluye una respuesta de resumen general (tipo MQCFT\_XR\_SUMMARY). La estructura de parámetros MQCACF\_RESPONSE\_Q\_MGR\_NAME identifica el gestor de colas que genera cada conjunto.

El conjunto de respuesta inicial tiene las siguientes estructuras de parámetro adicionales:




- MQIACF\_COMMAND\_INFO (MQCFIN). Los posibles valores de esta estructura son MQCMDI\_CMDSCOPE\_ACCEPTED o MQCMDI\_CMDSCOPE\_GENERATED.
- MQIACF\_CMDSCOPE\_Q\_MGR\_COUNT (MQCFIN). Esta estructura indica el número de gestores de colas a los que se envía el mandato.

### **Comprobación de autorización para mandatos PCF en IBM MQ**

Cuando se procesa un mandato PCF, se utiliza el *UserIdentifier* del descriptor de mensaje en el mensaje de mandato para las comprobaciones de autorización sobre objetos necesarias de IBM MQ. La comprobación de autorización se implementa de forma diferente en cada plataforma como se describe en este tema.

Las comprobaciones se realizan en el sistema en el que se está procesando el mandato; por lo tanto, este ID de usuario debe existir en el sistema de destino y tener las autorizaciones necesarias para procesar el mandato. Si el mensaje proviene de un sistema remoto, una forma de alcanzar el ID existente en el sistema de destino es tener un ID de usuario coincidente en los sistemas local y remoto.

**Nota:**  Para obtener más información sobre la comprobación de autorización en z/OS, consulte [Tarea 1: Identificar los parámetros del sistema z/OS](#).

### **IBM MQ for IBM i**



Para procesar cualquier mandato PCF, el ID de usuario debe tener autorización *dsp* para el objeto de IBM MQ en el sistema de destino.

Además, se realizan comprobaciones de autorización de objetos de IBM MQ para determinados mandatos PCF, como se muestra en la [Tabla 2 en la página 34](#).

En la mayoría de los casos, estas comprobaciones son las mismas comprobaciones que las comprobaciones realizadas por los mandatos CL de IBM MQ equivalentes emitidos en un sistema local. Consulte [Configuración de la seguridad en IBM i](#) para obtener más información sobre la correlación de autorizaciones de IBM MQ con autorizaciones del sistema IBM i, y los requisitos de autorización para los mandatos CL de IBM MQ. En [Seguridad a nivel de enlace mediante una salida de seguridad](#) se ofrece información detallada sobre la seguridad.

**Para procesar alguno de los mandatos siguientes**, el ID de usuario debe ser miembro del perfil de grupos QMQADM:

- Sondear canal
- Cambiar canal
- Copiar canal
- Crear canal
- Suprimir canal
- Restablecer canal
- Resolver canal
- Iniciar canal
- Detener canal
- Iniciar iniciador de canal
- Iniciar escucha de canal

### **IBM MQ for UNIX, Linux, and Windows**



Para procesar cualquier mandato PCF, el ID de usuario debe tener autorización *dsp* para el objeto de gestor de colas en el sistema de destino. Además, se realizan comprobaciones de autorización de objetos de IBM MQ para determinados mandatos PCF, como se muestra en la [Tabla 2 en la página 34](#).

**Para procesar cualquiera de los mandatos siguientes**, el ID de usuario debe pertenecer al grupo *mqm*.

**Nota:** Para Windows **únicamente**, el ID de usuario puede pertenecer al grupo *Administradores* o al grupo *mqm*.

- Cambiar canal
- Copiar canal
- Crear canal
- Suprimir canal
- Sondear canal
- Restablecer canal
- Iniciar canal
- Detener canal
- Iniciar iniciador de canal
- Iniciar escucha de canal
- Resolver canal
- Restablecer clúster
- Renovar clúster
- Suspender gestor de colas
- Reanudar gestor de colas

## Autorizaciones de objetos de IBM MQ para Multiplatforms



<i>Tabla 2. Autorizaciones de objetos</i>		
<b>Mandato</b>	<b>Autorización de objetos de IBM MQ</b>	<b>Autorización de clases (para tipo de objeto)</b>
Modificar información de autorización	dsp y chg	n/d
Cambiar canal	dsp y chg	n/d
Modificar escucha de canal	dsp y chg	n/d
Modificar canal de conexión de cliente	dsp y chg	n/d
Modificar lista de nombres	dsp y chg	n/d
Modificar proceso	dsp y chg	n/d
Modificar cola	dsp y chg	n/d
Cambiar gestor de colas	chg véase la nota 3 y la nota 5	n/d
Cambiar servicio	dsp y chg	n/d
Borrar cola	clr	n/d
Copiar información de autenticación	dsp	crt
Copiar información de autenticación (reemplazar) ver nota 1	de: dsp a: chg	crt
Copiar canal	dsp	crt

Tabla 2. Autorizaciones de objetos (continuación)

<b>Mandato</b>	<b>Autorización de objetos de IBM MQ</b>	<b>Autorización de clases (para tipo de objeto)</b>
Copiar canal (reemplazar) <i>ver nota 1</i>	de: dsp a: chg	crt
Copiar escucha de canal	dsp	crt
Copiar escucha de canal (reemplazar) <i>ver nota 1</i>	de: dsp a: chg	crt
Copiar canal de conexión de cliente	dsp	crt
Copiar canal de conexión de cliente (reemplazar) <i>ver nota 1</i>	de: dsp a: chg	crt
Copiar lista de nombres	dsp	crt
Copiar lista de nombres (reemplazar) <i>ver nota 1</i>	de: dsp a: dsp y chg	crt
Copiar proceso	dsp	crt
Copiar proceso (reemplazar) <i>ver nota 1</i>	de: dsp a: chg	crt
Copiar cola	dsp	crt
Copiar cola (reemplazar) <i>ver nota 1</i>	de: dsp a: dsp y chg	crt
Crear información de autenticación	(información de autenticación predeterminada del sistema) dsp	crt
Crear información de autenticación (reemplazar) <i>ver nota 1</i>	(información de autenticación predeterminada del sistema) dsp a: chg	crt
Crear canal	(canal predeterminado del sistema) dsp	crt
Crear canal (reemplazar) <i>ver nota 1</i>	(canal predeterminado del sistema) dsp a: chg	crt
Crear escucha de canal	(escucha predeterminado del sistema) dsp	crt
Crear escucha de canal (reemplazar) <i>ver nota 1</i>	(escucha predeterminado del sistema) dsp a: chg	crt
Crear canal de conexión de cliente	(canal predeterminado del sistema) dsp	crt
Crear canal de conexión de cliente (reemplazar) <i>ver nota 1</i>	(canal predeterminado del sistema) dsp a: chg	crt
Crear lista de nombres	(lista de nombres predeterminada del sistema) dsp	crt
Crear lista de nombres (reemplazar) <i>ver nota 1</i>	(lista de nombres predeterminada del sistema) dsp a: dsp y chg	crt
Crear proceso	(proceso predeterminado del sistema) dsp	crt
Crear proceso (reemplazar) <i>ver nota 1</i>	(proceso predeterminado del sistema) dsp a: chg	crt

Tabla 2. Autorizaciones de objetos (continuación)

Mandato	Autorización de objetos de IBM MQ	Autorización de clases (para tipo de objeto)
Crear cola	(cola predeterminada del sistema) dsp	crt
Crear cola (reemplazar) <i>ver nota 1</i>	(cola predeterminada del sistema) dsp a: dsp y chg	crt
Crear servicio	(cola predeterminada del sistema) dsp	crt
Crear servicio (reemplazar) <i>ver nota 1</i>	(cola predeterminada del sistema) dsp a: chg	crt
Suprimir información de autenticación	dsp y dlt	n/d
Suprimir registro de autorización	(objeto de gestor de colas) chg <i>ver nota 4</i>	<i>ver nota 4</i>
Suprimir canal	dsp y dlt	n/d
Suprimir escucha de canal	dsp y dlt	n/d
Suprimir canal de conexión de cliente	dsp y dlt	n/d
Suprimir lista de nombres	dsp y dlt	n/d
Suprimir proceso	dsp y dlt	n/d
Suprimir cola	dsp y dlt	n/d
Suprimir servicio	dsp y dlt	n/d
Consultar información de autenticación	dsp	n/d
Consultar registros de autorización	<i>ver nota 4</i>	<i>ver nota 4</i>
Consultar canal	dsp	n/d
Consultar escucha de canal	dsp	n/d
Consultar estado de canal (para <b>ChannelType</b> MQCHT_CLSSDR)	inq	n/d
Consultar canal de conexión de cliente	dsp	n/d
Consultar lista de nombres	dsp	n/d
Consultar proceso	dsp	n/d
Consultar cola	dsp	n/d
Consultar gestor de colas	<i>ver nota 3</i>	n/d
Consultar estado de la cola	dsp	n/d
Consultar servicio	dsp	n/d
Sondear canal	ctrl	n/d
Sondear gestor de colas	<i>ver nota 3</i>	n/d
Renovar gestor de colas	(objeto de gestor de colas) chg	n/d

Tabla 2. Autorizaciones de objetos (continuación)

Mandato	Autorización de objetos de IBM MQ	Autorización de clases (para tipo de objeto)
Renovar seguridad (para <b>SecurityType</b> MQSECTYPE_SSL)	(objeto de gestor de colas) chg	n/d
Restablecer canal	ctrlx	n/d
Restablecer gestor de colas	(objeto de gestor de colas) chg	n/d
Restablecer estadísticas de la cola	dsp y chg	n/d
Resolver canal	ctrlx	n/d
Establecer registro de autorización	(objeto de gestor de colas) chg ver nota 4	ver nota 4
Iniciar canal	ctrl	n/d
Detener canal	ctrl	n/d
Detener conexión	(objeto de gestor de colas) chg	n/d
Iniciar escucha	ctrl	n/d
Detener escucha	ctrl	n/d
Iniciar servicio	ctrl	n/d
Detener servicio	ctrl	n/d
Esc	ver nota 2	ver nota 2

**Notas:**

1. Este mandato se aplica si el objeto que debe reemplazarse ya existe, de lo contrario la comprobación de autorización es como para Crear o Copiar sin reemplazar.
2. La autorización necesaria se determina mediante el mandato MQSC definido por el texto de escape y es equivalente a uno de los mandatos anteriores.
3. Para procesar cualquier mandato PCF, el ID de usuario debe tener autorización dsp para el objeto de gestor de colas en el sistema de destino.
4. Este mandato PCF está autorizado a menos que el servidor de mandatos se haya iniciado con el parámetro -a. De forma predeterminada, el servidor de mandatos se inicia al iniciarse el gestor de colas, sin el parámetro -a. Para obtener más información, consulte [Referencia de formatos de mandatos programables](#).
5. Otorgar a un ID de usuario autorización chg para un gestor de colas concede la capacidad de establecer registros de autorización para todos los grupos y usuarios. Esta autorización no debe otorgarse a usuarios ni aplicaciones ordinarios.

IBM MQ también proporciona algunos puntos de salida de seguridad de canal para que pueda suministrar sus propios programas de salida de usuario para la comprobación de seguridad. Para obtener más información, consulte [Visualización de un canal](#).

**Multi Utilizar la MQAI para simplificar el uso de los PCF**

La interfaz de administración de IBM MQ (MQAI) es una interfaz de programación para IBM MQ que está disponible en AIX, IBM i, Linux, Solaris, y Windows. Realiza tareas de administración en un gestor de colas de IBM MQ mediante paquetes de datos para manejar las propiedades (o parámetros) de objetos de forma que es más fácil que utilizando los formatos de mandato programable (PCF).

La MQAI realiza tareas de administración en un gestor de colas mediante el uso de *paquetes de datos*. Los paquetes de datos permiten manejar las propiedades (o parámetros) de los objetos de forma más sencilla que utilizando mandatos PCF.

Las ventajas de utilizar la MQAI son las siguientes:

#### **Simplificar el uso de los mensajes PCF**

La MQAI es una forma más fácil de administrar IBM MQ. Si utiliza la MQAI, no tiene que escribir sus propios mensajes PCF. Esto evita los problemas asociados a las estructuras de datos complejas.

Para pasar parámetros en programas escritos utilizando llamadas MQI, el mensaje PCF debe contener el mandato y detalles de los datos de tipo entero o de serie de caracteres. Para crear esta configuración manualmente, tiene que añadir varias sentencias al programa para cada estructura y asignar espacio de memoria. Esta tarea puede resultar larga y laboriosa.

Los programas escritos utilizando la MQAI pasan parámetros en el paquete de datos adecuado y tan solo se requiere una sentencia para cada estructura. La utilización de paquetes de datos de la MQAI elimina la necesidad de manejar matrices y asignar almacenamiento y proporciona cierto grado de aislamiento de los detalles del PCF.

#### **Manejar las condiciones de error con más facilidad**

Es difícil obtener códigos de retorno de los mandatos PCF. La MQAI permite al programa manejar más fácilmente las condiciones de error.

#### **Intercambiar datos entre aplicaciones**

Los datos de aplicación se envían en formato PCF y los empaqueta y desempaqueta la MQAI. Si los datos del mensaje constan de enteros y series de caracteres, puede utilizar la MQAI para sacar partido de la conversión de datos incorporada de IBM MQ para datos PCF. Esto evita la necesidad de grabar salidas de conversión de datos.

Después de haber creado y llenado los paquetes de datos, puede enviar un mensaje de mandato de administración al servidor de mandatos de un gestor de colas, utilizando la llamada `mqExecute`. Esta llamada espera los mensajes de respuesta. La llamada `mqExecute` maneja el intercambio con el servidor de mandatos y devuelve respuestas en un *paquete de respuestas*.

### **Ejemplos de cómo utilizar la MQAI**

Los programas de ejemplo siguientes muestran el uso de la interfaz de administración de IBM MQ para realizar las diversas tareas:

- [`amqsaicq.c`](#): crear una cola local.
- [`amqsailem.c`](#): mostrar sucesos en la pantalla utilizando un supervisor de sucesos simple.
- [`amqsailq.c`](#): imprimir una lista de todas las colas locales y sus profundidades actuales.
- [`amqsaicl.c`](#): imprimir una lista de todos los canales y sus tipos.

### **Creación de una aplicación MQAI**

Para crear la aplicación utilizando la MQAI, debe enlazar con las mismas bibliotecas, tal como lo haría para IBM MQ. Para obtener información sobre cómo crear las aplicaciones de IBM MQ, consulte [Creación de una aplicación de procedimientos](#).

### **Consejos y sugerencias para configurar IBM MQ utilizando MQAI**

La MQAI utiliza los mensajes PCF para enviar mandatos de administración al servidor de mandatos, en lugar de tratar directamente con el propio servidor de mandatos. Encontrará consejos para configurar IBM MQ utilizando la MQAI en [“Consejos y sugerencias para utilizar MQAI para configurar IBM MQ”](#) en la [página 39](#).

#### **Referencia relacionada**

[Referencia de la interfaz de administración de IBM MQ](#)

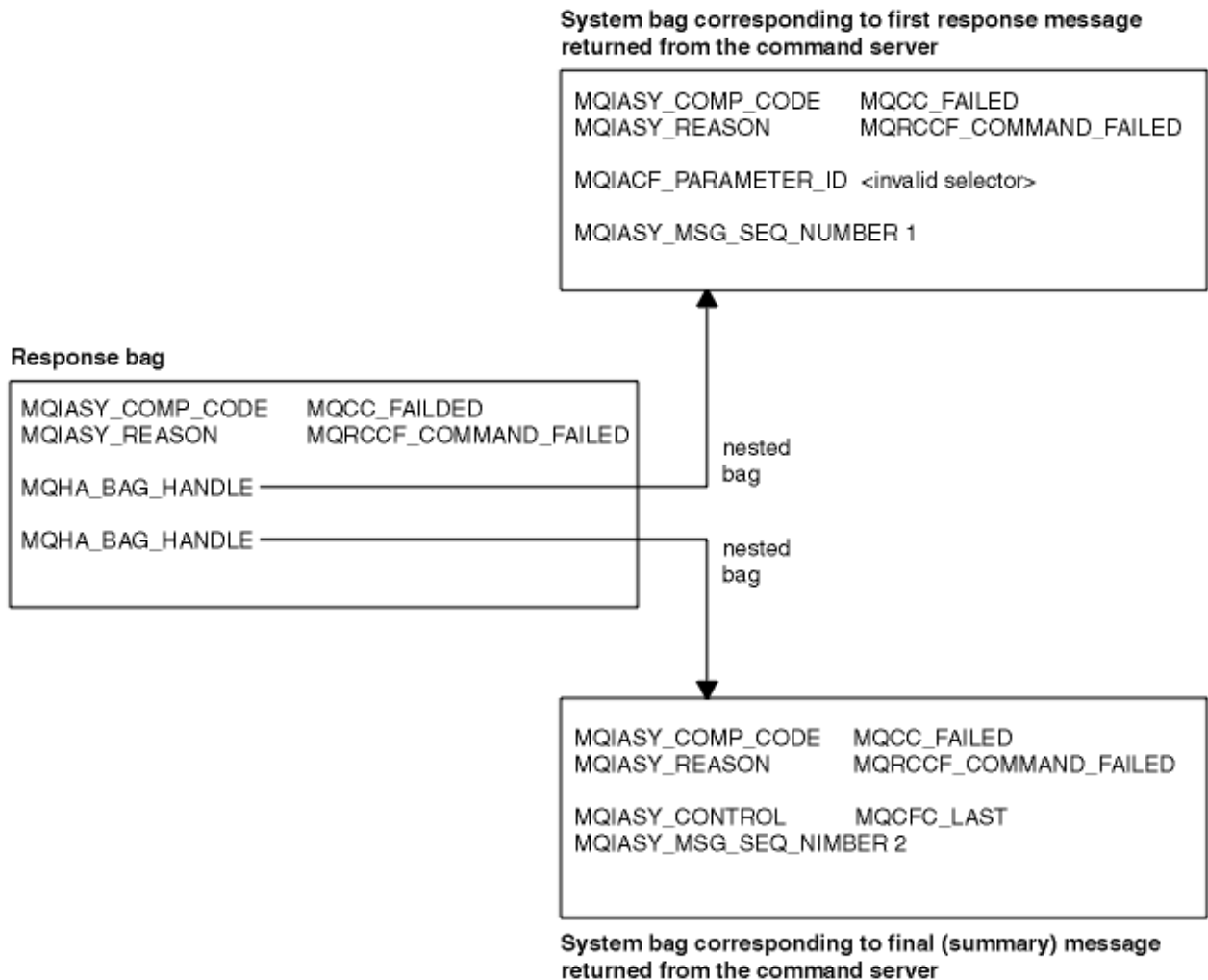
## Consejos y sugerencias para utilizar MQAI para configurar IBM MQ

La Interfaz de administración de IBM MQ (MQAI) utiliza los mensajes PCF para enviar mandatos de administración al servidor de mandatos, en lugar de tratar directamente con el propio servidor de mandatos. A continuación se proporcionan algunas sugerencias para configurar IBM MQ utilizando la MQAI.

- Las series de caracteres en IBM MQ se rellenan con blancos hasta una longitud fija. Utilizando C, las series terminadas en nulo normalmente pueden proporcionarse como parámetros de entrada a las interfaces de programación de IBM MQ.
- Para borrar el valor de un atributo de serie, establézcalo en un único blanco en lugar de una serie vacía.
- Piense con antelación los atributos que desea cambiar y realice consultas únicamente sobre los mismos.
- Determinados atributos no pueden cambiarse, por ejemplo un nombre de cola o un tipo de canal. Asegúrese de que intenta cambiar únicamente los atributos que se pueden modificar. Consulte la lista de parámetros necesarios y opcionales para el objeto de cambio PCF específico. Consulte [Definiciones de los formatos de mandatos programables](#).
- Si una llamada MQAI falla, algunos detalles del error se devuelven al paquete de respuesta. Se pueden encontrar más detalles en un paquete anidado al que puede acceder el selector MQHA\_BAG\_HANDLE. Por ejemplo, si una llamada mqExecute falla con un código de razón MQRCCF\_COMMAND\_FAILED, esta información se devuelve en el paquete de respuesta. Una causa posible de este código de razón es que un selector especificado no es válido para el tipo de mensaje de mandato y este detalle de la información se encuentra en un paquete anidado al que puede acceder un manejador de paquete.

Para obtener más información sobre MQExecute, consulte [“Envío de mandatos de administración al servidor de mandatos qm utilizando la llamada mqExecute”](#) en la [página 73](#)

El siguiente diagrama muestra este escenario:



## Multi **Temas de la interfaz de administración de IBM MQ avanzados**

Información sobre la indexación, conversión de datos y uso del descriptor de mensaje

### **Indexación**

Los índices se utilizan cuando se sustituyen o eliminan los elementos de datos existentes de un paquete para conservar el orden de inserción.

### **Conversión de datos**

Las series contenidas en un paquete de datos MQAI (interfaz de administración de IBM MQ) pueden estar en distintos juegos de caracteres codificados y pueden convertirse utilizando la llamada `mqSetInteger`.

### **Uso del descriptor de mensaje**

La interfaz de administración de IBM MQ genera un descriptor de mensaje que se establece en un valor inicial cuando se crea el paquete de datos.

## Multi **Indexación en la interfaz de administración de IBM MQ**

Los índices se utilizan cuando se sustituyen o eliminan los elementos de datos existentes de un paquete. Existen tres tipos de indexación, que permite recuperar fácilmente los elementos de datos.

Cada selector y valor dentro de un elemento de datos de un paquete tiene tres números de asociados:

- El índice relativo a otros elementos que tienen el mismo selector.
- El índice relativo a la categoría de selector (usuario o sistema) al que pertenece el elemento.
- El índice relativo a todos los elementos de datos del paquete (usuario y sistema).



Esto permite la indexación según los selectores de usuario, los selectores del sistema, o ambos como se muestra en la [Figura 3](#) en la página 41.

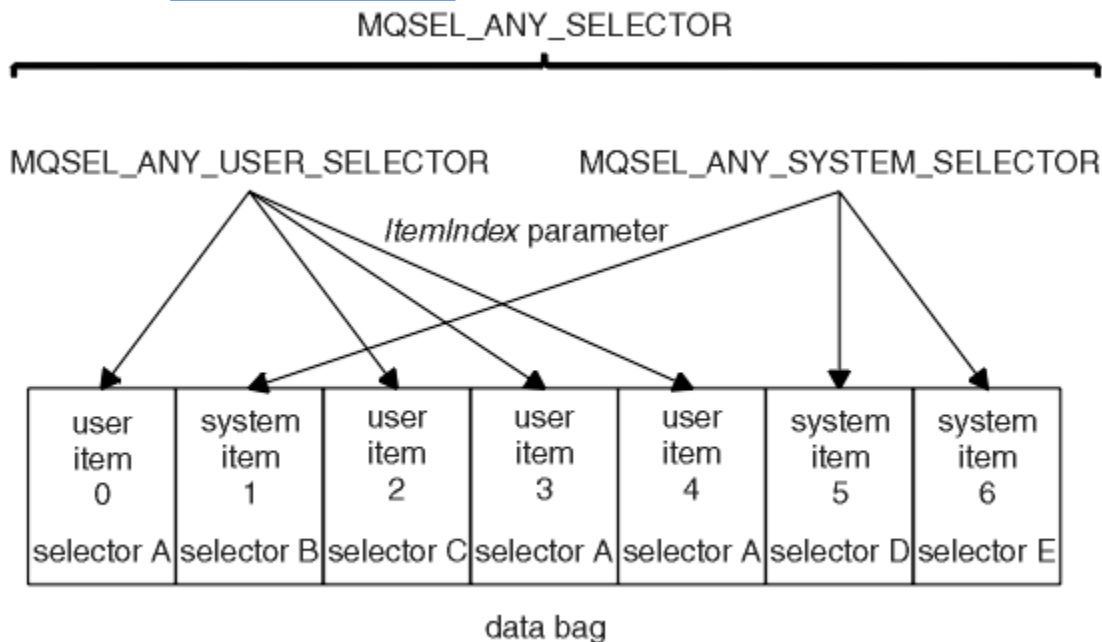


Figura 3. Indexación

En la [Figura 3](#) en la [página 41](#), se puede hacer referencia al elemento de usuario 3 (selector A) mediante los pares de índice siguientes:

- selector A (ItemIndex 1)
- MQSEL\_ANY\_USER\_SELECTOR (ItemIndex 2)
- MQSEL\_ANY\_SELECTOR (itemIndex 3)

El índice tiene base cero como una matriz en c; si hay 'n' apariciones, el índice oscila entre cero y 'n-1', sin espacios.

Los índices se utilizan cuando se sustituyen o eliminan los elementos de datos existentes de un paquete. Cuando se utilizan de esta manera, el orden de inserción se conserva pero los índices de otros elementos de datos pueden verse afectados. Para obtener ejemplos de esto, consulte [Cambiar información dentro de un paquete](#) y [Supresión de elementos de datos](#).

Los tres tipos de indexación permiten una recuperación fácil de elementos de datos. Por ejemplo, si hay tres instancias de un selector concreto en un paquete, la llamada mqCountItems puede contar el número de instancias de ese selector y las llamadas mqInquire\* pueden especificar tanto el selector como el índice para consultar solamente los valores. Esto es útil para los atributos que pueden tener una lista de valores como algunas de las salidas en los canales.

### **Multi** Proceso de conversión de datos en la interfaz de administración de IBM MQ

Las series contenidas en un paquete de datos MQAI pueden estar en diversos juegos de caracteres codificados. Estas series se pueden convertir utilizando la llamada mqSetInteger.

Al igual que los mensajes PCF, las series contenidas en un paquete de datos MQAI pueden estar en diversos juegos de caracteres codificados. Normalmente, todas las series en un mensaje PCF están en el mismo juego de caracteres codificado; es decir, el mismo conjunto que el gestor de colas.

Cada elemento de serie del paquete de datos contiene dos valores; la propia serie y el CCSID. La serie que se añade al paquete se obtiene del parámetro **Buffer** de la llamada mqAddString o mqSetString. El CCSID se obtiene del elemento del sistema que contiene un selector de MQIASY\_CODED\_CHAR\_SET\_ID. Esto se conoce como el *CCSID de paquete* y se puede cambiar utilizando la llamada mqSetInteger.

Al consultar el valor de una serie incluida en un paquete de datos, el CCSID es un parámetro de salida de la llamada.

Tabla 3 en la página 42 muestra las normas aplicadas al convertir los paquetes de datos en mensajes y viceversa:

<i>Tabla 3. Proceso de CCSID</i>			
<b>Llamada de MQAI</b>	<b>CCSID</b>	<b>Entrada para llamada</b>	<b>Salida para llamada</b>
<a href="#">mqBagToBuffer</a>	CCSID incorrecto ( <u>1</u> )	Se ignora	Sin modificar
<a href="#">mqBagToBuffer</a>	CCSID de series del paquete	Se utiliza	Sin modificar
<a href="#">mqBagToBuffer</a>	CCSID de series del almacenamiento intermedio	No aplicable	Se copia de los CCSID de series del paquete
<a href="#">mqBufferToBag</a>	CCSID incorrecto ( <u>1</u> )	Se ignora	Sin modificar
<a href="#">mqBufferToBag</a>	CCSID de series del almacenamiento intermedio	Se utiliza	Sin modificar
<a href="#">mqBufferToBagmqBufferToBag</a>	CCSID de series del paquete	No aplicable	Se copia de los CCSID de serie del almacenamiento intermedio
<a href="#">mqPutBag</a>	CCSID de MQMD	Se utiliza	Sin modificar ( <u>2</u> )
<a href="#">mqPutBag</a>	CCSID incorrecto ( <u>1</u> )	Se ignora	Sin modificar
<a href="#">mqPutBag</a>	CCSID de series del paquete	Se utiliza	Sin modificar
<a href="#">mqPutBag</a>	CCSID de series del mensaje enviado	No aplicable	Se copia de los CCSID de series del paquete
<a href="#">mqGetBag</a>	CCSID de MQMD	Se utiliza para la conversión de datos de mensaje	Establecido en CCSID de datos devueltos ( <u>3</u> )
<a href="#">mqGetBag</a>	CCSID incorrecto ( <u>1</u> )	Se ignora	Sin modificar
<a href="#">mqGetBag</a>	CCSID de series de mensaje	Se utiliza	Sin modificar
<a href="#">mqGetBag</a>	CCSID de series del paquete	No aplicable	Se copia de los CCSID de series del mensaje
<a href="#">mqExecute</a>	CCSID de paquete de solicitud	Utilizado para MQMD del mensaje de solicitud ( <u>4</u> )	Sin modificar
<a href="#">mqExecute</a>	CCSID de paquete de respuesta	Se utiliza para la conversión de datos del mensaje de respuesta ( <u>4</u> )	Establecido en CCSID de datos devueltos ( <u>3</u> )
<a href="#">mqExecute</a>	CCSID de series de paquete de solicitud	Se utiliza para el mensaje de solicitud	Sin modificar

<i>Tabla 3. Proceso de CCSID (continuación)</i>			
<b>Llamada de MQAI</b>	<b>CCSID</b>	<b>Entrada para llamada</b>	<b>Salida para llamada</b>
mqExecute	CCSID de series en paquete de respuesta	No aplicable	Se copia de los CCSID de serie del mensaje de respuesta

**Notas:**

1. El CCSID de paquete es el elemento del sistema con el selector MQIASY\_CODED\_CHAR\_SET\_ID.
2. MQCCSI\_Q\_MGR se cambia por el CCSID del gestor de colas real.
3. Si se solicita la conversión de datos, el CCSID de datos devuelto es el mismo que el valor de salida. Si no se solicita la conversión de datos, el CCSID de datos devueltos es el mismo que el valor de mensaje. Tenga en cuenta que se devuelve ningún mensaje si se solicita la conversión de datos pero falla.
4. Si el CCSID es MQCCSI\_DEFAULT, se utilizará el CCSID del gestor de colas.

**Conceptos relacionados**

“Conversión de datos entre juegos de caracteres codificados” en la página 215

El gestor de colas puede convertir datos de mensaje en formatos definidos de IBM MQ (también conocidos como formatos incorporados) de un juego de caracteres codificado a otro, siempre que ambos juegos de caracteres estén relacionados con un único idioma o un grupo de idiomas similares.

“El archivo ccsid\_part2.tbl” en la página 217

El archivo ccsid\_part2.tbl se utiliza para proporcionar información CCSID adicional. El archivo ccsid\_part2.tbl sustituye el archivo ccsid.tbl que se ha utilizado antes de IBM MQ 9.0.

**Multi** **Uso del descriptor de mensaje en la interfaz de administración de IBM MQ**

El descriptor de mensaje que se la interfaz de administración de IBM MQ genera se establece en un valor inicial cuando se crea el paquete de datos.

El tipo de mandato PCF se obtiene del elemento del sistema con el selector MQIASY\_TYPE. Al crear el paquete de datos, el valor inicial de este elemento se establece en función del tipo de paquete que se crea:

<i>Tabla 4. Tipo de mandato PCF</i>	
<b>Tipo de paquete</b>	<b>Valor inicial del elemento MQIASY_TYPE</b>
MQCBO_ADMIN_BAG	MQCFT_COMMAND
MQCBO_COMMAND_BAG	MQCFT_COMMAND
MQCBO_*	MQCFT_USER

Cuando la MQAI genera un descriptor de mensaje, los valores utilizados en los parámetros **Format** y **MsgType** dependen del valor del elemento del sistema con el selector MQIASY\_TYPE tal como se muestra en la [Tabla 4](#) en la [página 43](#).

<i>Tabla 5. Formato y parámetros MsgType del MQMD</i>		
<b>Tipo de mandato PCF</b>	<b>Formato</b>	<b>MsgType</b>
MQCFT_COMMAND	MQFMT_ADMIN	MQMT_REQUEST
MQCFT_REPORT	MQFMT_ADMIN	MQMT_REPORT
MQCFT_RESPONSE	MQFMT_ADMIN	MQMT_REPLY
MQCFT_TRACE_ROUTE	MQFMT_ADMIN	MQMT_DATAGRAM
MQCFT_EVENT	MQFMT_EVENT	MQMT_DATAGRAM

Tabla 5. Formato y parámetros <i>MsgType</i> del MQMD (continuación)		
Tipo de mandato PCF	Formato	MsgType
MQCFT_*	MQFMT_PCF	MQMT_DATAGRAM

Tabla 5 en la página 43 muestra que si crea un paquete de administración o un paquete de mandatos, el *Format* del descriptor de mensaje es MQFMT\_ADMIN y el *MsgType* es MQMT\_REQUEST. Esto resulta adecuado para un mensaje de petición PCF que se envía al servidor de mandatos cuando se espera una respuesta.

Otros parámetros del descriptor de mensaje adoptan los valores que se muestran en la [Tabla 6](#) en la [página 44](#).

Tabla 6. Valores de descriptor de mensaje	
Parámetro	Valor
<i>StrucId</i>	MQMD_STRUC_ID
<i>Version</i>	MQMD_VERSION_1
<i>Report</i>	MQRO_NONE
<i>MsgType</i>	consulte <a href="#">Tabla 5</a> en la <a href="#">página 43</a>
<i>Expiry</i>	30 segundos (vea <a href="#">“1”</a> en la <a href="#">página 44</a> )
<i>Feedback</i>	MQFB_NONE
<i>Encoding</i>	MQENC_NATIVE
<i>CodedCharSetId</i>	depende del CCSID del paquete (nota <a href="#">“2”</a> en la <a href="#">página 44</a> )
<i>Format</i>	consulte <a href="#">Tabla 5</a> en la <a href="#">página 43</a>
<i>Priority</i>	MQPRI_PRIORITY_AS_Q_DEF
<i>Persistence</i>	MQPER_NOT_PERSISTENT
<i>MsgId</i>	MQMI_NONE
<i>CorrelId</i>	MQCI_NONE
<i>BackoutCount</i>	0
<i>ReplyToQ</i>	vea la nota <a href="#">“3”</a> en la <a href="#">página 44</a>
<i>ReplyToQMGr</i>	en blanco

#### Notas:

1. Este valor puede alterarse en la llamada `mqExecute` utilizando el parámetro **OptionsBag**. Para obtener información sobre esto, consulte [mqExecute](#).
2. Consulte [“Proceso de conversión de datos en la interfaz de administración de IBM MQ”](#) en la [página 41](#).
3. El nombre de la cola de respuesta especificada por el usuario o la cola dinámica temporal generada por la MQAI para los mensajes del tipo MQMT\_REQUEST. De lo contrario, espacio en blanco.

### Programa C de ejemplo para crear una cola local (amqsaicq.c)

El programa C de ejemplo `amqsaicq.c` crea una cola local utilizando MQAI.

```

/*****

```

```

/*                                                                    */
/* Program name: AMQSAICQ.C                                           */
/*                                                                    */
/* Description:  Sample C program to create a local queue using the   */
/*              IBM MQ Administration Interface (MQAI).               */
/*                                                                    */
/* Statement:    Licensed Materials - Property of IBM                 */
/*                                                                    */
/*              84H2000, 5765-B73                                       */
/*              84H2001, 5639-B42                                       */
/*              84H2002, 5765-B74                                       */
/*              84H2003, 5765-B75                                       */
/*              84H2004, 5639-B43                                       */
/*                                                                    */
/*              (C) Copyright IBM Corp. 1999, 2024.                   */
/*                                                                    */
/*****                                                                    */
/*                                                                    */
/* Function:                                                                    */
/* AMQSAICQ is a sample C program that creates a local queue and is an */
/* example of the use of the mqExecute call.                               */
/*                                                                    */
/* - The name of the queue to be created is a parameter to the program. */
/*                                                                    */
/* - A PCF command is built by placing items into an MQAI bag.         */
/*   These are:-                                                         */
/*     - The name of the queue                                           */
/*     - The type of queue required, which, in this case, is local.    */
/*                                                                    */
/* - The mqExecute call is executed with the command MQCMD_CREATE_Q.   */
/*   The call generates the correct PCF structure.                     */
/*   The call receives the reply from the command server and formats into */
/*   the response bag.                                                  */
/*                                                                    */
/* - The completion code from the mqExecute call is checked and if there */
/*   is a failure from the command server then the code returned by the */
/*   command server is retrieved from the system bag that is           */
/*   embedded in the response bag to the mqExecute call.               */
/*                                                                    */
/* Note: The command server must be running.                            */
/*                                                                    */
/*****                                                                    */
/*                                                                    */
/* AMQSAICQ has 2 parameters - the name of the local queue to be created */
/* - the queue manager name (optional)                                   */
/*                                                                    */
/*****                                                                    */
/* Includes                                                                    */
/*****                                                                    */
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>

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

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

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

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

```

```

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

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

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

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

}

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

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

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

```

```

return;

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

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

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

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

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

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

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

```

```

        reason);
    printf("Error returned by the command server: Completion code = %d :
           Reason = %d\n", mqExecuteCC, mqExecuteRC);
}
}
/*****
/* Delete the command bag if successfully created. */
/*****
if (commandBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&commandBag, &compCode, &reason);
    CheckCallResult("Delete the command bag", compCode, reason);
}

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

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

```

## **Multi** Programa C de ejemplo para visualizar sucesos utilizando un supervisor de sucesos (amqsaie.m.c)

El programa C de ejemplo amqsaie.m.c muestra un supervisor de sucesos básico utilizando MQAI.

```

*****/
/*
/* Program name: AMQSAIEM.C */
/*
/* Description: Sample C program to demonstrate a basic event monitor */
/* using the IBM MQ Admin Interface (MQAI). */
/* Licensed Materials - Property of IBM */
/*
/*
/* 63H9336 */
/* (c) Copyright IBM Corp. 1999, 2024. All Rights Reserved. */
/*
/*
/* US Government Users Restricted Rights - Use, duplication or */
/* disclosure restricted by GSA ADP Schedule Contract with */
/* IBM Corp. */
/*****
/*
/* Function: */
/* AMQSAIEM is a sample C program that demonstrates how to write a simple */
/* event monitor using the mqGetBag call and other MQAI calls. */
/*
/*
/* The name of the event queue to be monitored is passed as a parameter */
/* to the program. This would usually be one of the system event queues:- */
/* SYSTEM.ADMIN.QMGR.EVENT Queue Manager events */
/*

```



```

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

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

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

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

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

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

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

/*****

```

```

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

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

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

return 0;
}

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

/*****
/*
/* Function: GetQEvents */
/*
/*****
/*
/* Input Parameters: Handle to the queue manager */
/* Name of the event queue to be monitored */
/*
/* Output Parameters: None */
/*
/* Logic: Open the event queue. */
/* Get a message off the event queue and format the message into */
/* a bag. */
/* A real event monitor would need to be programmed to deal with */
/* each type of event that it receives from the queue. This is */
/* outside the scope of this sample, so instead, the contents of */
/* the bag are printed. */
/* The program waits for 30 seconds for an event message and then */
/* terminates if no more messages are available. */
/*
/*****
void GetQEvents(MQHCONN hConn, MQCHAR *qName)
{
    MQLONG openReason; /* MQOPEN reason code */

```

```

MQLONG reason; /* reason code */
MQLONG compCode; /* completion code */
MQHOBJ eventQueue; /* handle to event queue */

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

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

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

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

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

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

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

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

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

/*****
/* Event message read - Print the contents of the event bag */
*****/
else
{
    if ( PrintBag(eventBag) )

```

```

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

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

} /* end of GetQEvents */

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

int PrintBag(MQHBAG dataBag)
{
    int errors;

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

    return errors;
}

/*****
/*
/* Function: PrintBagContents
/*
/*****
/*
/* Input Parameters: Bag Handle
/*
/* Indentation level of bag
/*
/*
/* Output Parameters: None
/*
/*
/* Returns: Number of errors found
/*
/*
/* Logic: Count the number of items in the bag
/*
/* Obtain selector and item type for each item in the bag.
/*
/* Obtain the value of the item depending on item type and display the
/*
/* index of the item, the selector and the value.
/*
/* If the item is an embedded bag handle then call this function again
/*
/* to print the contents of the embedded bag increasing the
/*
/* indentation level.
/*
/*****
int PrintBagContents(MQHBAG dataBag, int indent)
{
    /*****
    /* Definitions
    /*****
    /*****
    #define LENGTH 500 /* Max length of string to be read*/
    #define INDENT 4 /* Number of spaces to indent */
    /* embedded bag display */

```

```

/*****
/* Variables
/*****
MQLONG itemCount; /* Number of items in the bag */
MQLONG itemType; /* Type of the item */
int i; /* Index of item in the bag */
MQCHAR stringVal[LENGTH+1]; /* Value if item is a string */
MQBYTE byteStringVal[LENGTH]; /* Value if item is a byte string */
MQLONG stringLength; /* Length of string value */
MQLONG ccsid; /* CCSID of string value */
MQINT32 iValue; /* Value if item is an integer */
MQINT64 i64Value; /* Value if item is a 64-bit
/* integer
MQLONG selector; /* Selector of item
MQHBAG bagHandle; /* Value if item is a bag handle */
MQLONG reason; /* reason code
MQLONG compCode; /* completion code
MQLONG trimLength; /* Length of string to be trimmed
int errors = 0; /* Count of errors found
char blanks[] = " /* Blank string used to
/* indent display

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

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

/*****
/* If no errors found, display each item in the bag
/*****
if (!errors)
{
    for (i = 0; i < itemCount; i++)
    {

/*****
/* First inquire the type of the item for each item in the bag
/*****
mqInquireItemInfo(dataBag, /* Bag handle
/*
MQSEL_ANY_SELECTOR, /* Item can have any selector*/
i, /* Index position in the bag */
&selector, /* Actual value of selector
/* returned by call
/*
&itemType, /* Actual type of item
/* returned by call
/*
&compCode, /* Completion code
/*
&reason); /* Reason Code
/*

if (compCode != MQCC_OK)
    errors++;

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

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

```

```

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

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

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

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

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

/*****
/* The call can return a warning if the string is too long for */
/* the output buffer and has been truncated, so only check */
/* explicitly for call failure. */
/*****
if (compCode == MQCC_FAILED)
errors++;

```

```

else
{
    printf("%.s %-2d    %-4d    X'",
           indent, blanks, i, selector);

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

    printf("\n");
}
break;

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

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

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

```

## Multi Programa C de ejemplo para realizar consultas sobre objetos de canal (amqsaicl.c)

El programa C de ejemplo amqsaicl.c consulta los objetos de canal utilizando la MQAI.

```

/*****
/*
/* Program name: AMQSAICL.C
/*
/* Description: Sample C program to inquire channel objects
/*              using the IBM MQ Administration Interface (MQAI)
/*
/* <N_OCO_COPYRIGHT>
/* Licensed Materials - Property of IBM
/*
/* 63H9336
/* (c) Copyright IBM Corp. 2008, 2024. All Rights Reserved.
/*
/* US Government Users Restricted Rights - Use, duplication or
/* disclosure restricted by GSA ADP Schedule Contract with
/* IBM Corp.
/* <NOC_COPYRIGHT>
*****/
/*
/* Function:
/* AMQSAICL is a sample C program that demonstrates how to inquire
/* attributes of the local queue manager using the MQAI interface. In
/* particular, it inquires all channels and their types.
/*
/* - A PCF command is built from items placed into an MQAI administration
/* bag.
/*

```

```

/*      These are:-
/*      - The generic channel name "*"
/*      - The attributes to be inquired. In this sample we just want
/*      name and type attributes
/*
/*      - The mqExecute MQCMD_INQUIRE_CHANNEL call is executed.
/*      The call generates the correct PCF structure.
/*      The default options to the call are used so that the command is sent
/*      to the SYSTEM.ADMIN.COMMAND.QUEUE.
/*      The reply from the command server is placed on a temporary dynamic
/*      queue.
/*      The reply from the MQCMD_INQUIRE_CHANNEL is read from the
/*      temporary queue and formatted into the response bag.
/*
/*      - The completion code from the mqExecute call is checked and if there
/*      is a failure from the command server, then the code returned by the
/*      command server is retrieved from the system bag that has been
/*      embedded in the response bag to the mqExecute call.
/*
/* Note: The command server must be running.
/*
/*****
/* AMQSAICL has 2 parameter - the queue manager name (optional)
/*      - output file (optional) default varies
/*****

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

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

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

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

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

```



```

"sdr      ", /* MQCHT_SENDER */
"svr      ", /* MQCHT_SERVER */
"rcvr     ", /* MQCHT_RECEIVER */
"rqstr    ", /* MQCHT_REQUESTER */
"all      ", /* MQCHT_ALL */
"cltconn  ", /* MQCHT_CLNTCONN */
"svrcn    ", /* MQCHT_SVRCONN */
"cluscvr  ", /* MQCHT_CLUSRCVR */
"cluscdr  "  /* MQCHT_CLUSSDR */
};
#endif

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

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

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

#endif

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

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

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

/*****
/* Report the reason and stop if the connection failed. */
/*****

```

```

/*****
if (compCode == MQCC_FAILED)
{
    CheckCallResult("Queue Manager connection", compCode, connReason);
    exit( (int)connReason);
}

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

if(outfp == NULL)
{
    printf("Could not open output file.\n");
    goto MOD_EXIT;
}

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

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

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

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

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

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

/*****
/* Check the command server is started. If not exit. */
/*****
if (reason == MQRC_CMD_SERVER_NOT_AVAILABLE)
{
    printf("Please start the command server: <strmqcsv QMgrName="">\n");
    goto MOD_EXIT;
}

/*****

```

```

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

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

/*****
/* Get the channel name out of the channel attributes bag */
/*****
mqInquireString(cAttrsBag, MQCACH_CHANNEL_NAME, 0, MQ_OBJECT_NAME_LENGTH,
ch1Name, &ch1NameLength, NULL, &compCode, &reason);
CheckCallResult("Get channel name", compCode, reason);

/*****
/* Get the channel type out of the channel attributes bag */
/*****
mqInquireInteger(cAttrsBag, MQIACH_CHANNEL_TYPE, MQIND_NONE, &ch1Type,
&compCode, &reason);
CheckCallResult("Get type", compCode, reason);

/*****
/* Use mqTrim to prepare the channel name for printing. */
/* Print the result. */
/*****
mqTrim(MQ_CHANNEL_NAME_LENGTH, ch1Name, ch1Name, &compCode, &reason);
sprintf(OutputBuffer, "%-20s%-9s", ch1Name, Ch1Type2String(ch1Type));
WRITEOUTFILE(outfp,OutputBuffer,29)
}
}
else /* Failed mqExecute */
{
printf("Call to get channel attributes failed: Cc = %ld : Rc = %ld\n",
compCode, reason);
/*****
/* If the command fails get the system bag handle out of the mqexecute */
/* response bag. This bag contains the reason from the command server */
/* why the command failed. */
/*****
if (reason == MQRCCF_COMMAND_FAILED)
{
mqInquireBag(responseBag, MQHA_BAG_HANDLE, 0, &errorBag,
&compCode, &reason);
CheckCallResult("Get the result bag handle", compCode, reason);

/*****
/* Get the completion code and reason code, returned by the command */
/* server, from the embedded error bag. */
/*****
mqInquireInteger(errorBag, MQIASY_COMP_CODE, MQIND_NONE, &mqExecuteCC,
&compCode, &reason );
CheckCallResult("Get the completion code from the result bag",
compCode, reason);
mqInquireInteger(errorBag, MQIASY_REASON, MQIND_NONE, &mqExecuteRC,
&compCode, &reason);
CheckCallResult("Get the reason code from the result bag",
compCode, reason);
printf("Error returned by the command server: Cc = %ld : Rc = %ld\n",
mqExecuteCC, mqExecuteRC);
}
}
}
MOD_EXIT:
/*****

```

```

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

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

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

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

return 0;
}

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

```

## **Multi** Programa C de ejemplo para realizar consultas sobre colas e imprimir información (amqsailq.c)

El programa C de ejemplo amqsailq.c indaga la profundidad actual de las colas locales utilizando MQAI.

```

/*****
/*
/* Program name: AMQSAILQ.C */
/*
/* Description: Sample C program to inquire the current depth of the local */
/* queues using the IBM MQ Administration Interface (MQAI) */
/*
/* Statement: Licensed Materials - Property of IBM */
/*
/* 84H2000, 5765-B73 */
/* 84H2001, 5639-B42 */
/* 84H2002, 5765-B74 */
/* 84H2003, 5765-B75 */

```

```

/*          84H2004, 5639-B43          */
/*          (C) Copyright IBM Corp. 1999, 2024.          */
/*          ****          */
/* Function:          */
/* AMQSAILQ is a sample C program that demonstrates how to inquire          */
/* attributes of the local queue manager using the MQAI interface. In          */
/* particular, it inquires the current depths of all the local queues.          */
/*          */
/* - A PCF command is built by placing items into an MQAI administration          */
/* bag.          */
/* These are:-          */
/* - The generic queue name "*"          */
/* - The type of queue required. In this sample we want to          */
/* inquire local queues.          */
/* - The attribute to be inquired. In this sample we want the          */
/* current depths.          */
/*          */
/* - The mqExecute call is executed with the command MQCMD_INQUIRE_Q.          */
/* The call generates the correct PCF structure.          */
/* The default options to the call are used so that the command is sent          */
/* to the SYSTEM.ADMIN.COMMAND.QUEUE.          */
/* The reply from the command server is placed on a temporary dynamic          */
/* queue.          */
/* The reply from the MQCMD_INQUIRE_Q command is read from the          */
/* temporary queue and formatted into the response bag.          */
/*          */
/* - The completion code from the mqExecute call is checked and if there          */
/* is a failure from the command server, then the code returned by          */
/* command server is retrieved from the system bag that has been          */
/* embedded in the response bag to the mqExecute call.          */
/*          */
/* - If the call is successful, the depth of each local queue is placed          */
/* in system bags embedded in the response bag of the mqExecute call.          */
/* The name and depth of each queue is obtained from each of the bags          */
/* and the result displayed on the screen.          */
/*          */
/* Note: The command server must be running.          */
/*          ****          */
/* AMQSAILQ has 1 parameter - the queue manager name (optional)          */
/*          ****          */
/*          ****          */
/* Includes          */
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>

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

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

/*          ****          */
/* Function: main          */
/*          ****          */
int main(int argc, char *argv[])
{
    /*          ****          */
    /* MQAI variables          */
    /*          ****          */
    MQHCONN hConn;          /* handle to IBM MQ connection          */
    MQCHAR qmName[MQ_Q_MGR_NAME_LENGTH+1]=""; /* default QMgr name          */
    MQLONG reason;          /* reason code          */
    MQLONG connReason;          /* MQCONN reason code          */
    MQLONG compCode;          /* completion code          */
    MQHBAG adminBag = MQHB_UNUSABLE_HBAG; /* admin bag for mqExecute          */
    MQHBAG responseBag = MQHB_UNUSABLE_HBAG; /* response bag for mqExecute          */
    MQHBAG qAttrsBag;          /* bag containing q attributes          */
    MQHBAG errorBag;          /* bag containing cmd server error          */
    MQLONG mqExecuteCC;          /* mqExecute completion code          */

```

```

MQLONG mqExecuteRC;          /* mqExecute reason code          */
MQLONG qNameLength;         /* Actual length of q name        */
MQLONG qDepth;             /* depth of queue                 */
MQLONG i;                  /* loop counter                   */
MQLONG numberOfBags;       /* number of bags in response bag */
MQCHAR qName[MQ_Q_NAME_LENGTH+1]; /* name of queue extracted from bag*/

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

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

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

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

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

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

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

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

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

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

```

```

    exit(98);
}

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

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

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

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

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

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

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

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

```

```

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

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

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

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

```

## Multi Paquetes de datos y MQAI

Un paquete de datos es una manera de manejar propiedades o parámetros de objetos utilizando la Interfaz de administración de IBM MQ (MQAI).

### Paquetes de datos

- El paquete de datos contiene cero o más *elementos de datos*. Estos elementos de datos se ordenan dentro del paquete cuando se colocan en el paquete. Esto se denomina el *orden de inserción*. Cada elemento de datos contiene un *selector* que identifica el elemento de datos y un *valor* de ese elemento de datos que puede ser un entero, un entero de 64 bits, un filtro de enteros, una serie, un filtro de texto, una serie de bytes, un filtro de serie de bytes, o un manejador de otro paquete. Los elementos de datos se describen en detalle en [“Tipos de elemento de datos disponibles en la MQAI” en la página 67](#)

Existen dos tipos de selector; *selectores de usuario* y *selectores de sistema*. Éstos se describen en la sección [Selectores de MQAI](#). Los selectores normalmente son exclusivos, pero es posible tener varios valores para el mismo selector. En este caso, un *índice* identifica la aparición concreta del selector que



es necesario. Los índices se describen en [“Indexación en la interfaz de administración de IBM MQ”](#) en la página 40.

Una jerarquía de estos conceptos se muestra en la [Figura 1](#).

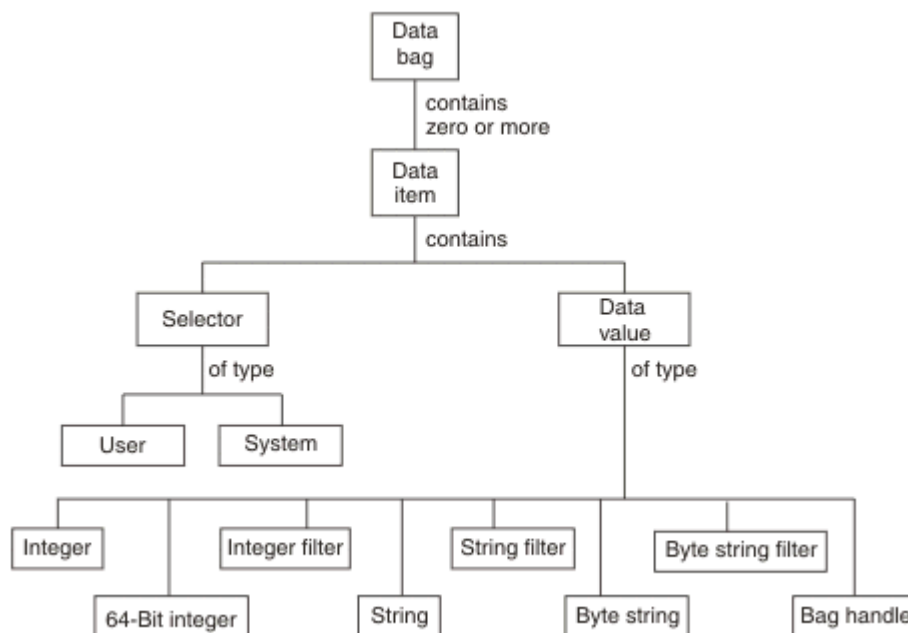


Figura 4. Jerarquía de los conceptos de MQAI

La jerarquía se ha explicado en un párrafo anterior.

## Tipos de paquetes de datos

Puede elegir el tipo de paquete de datos que desea crear según la tarea que desee realizar:

### paquete de usuario

Un paquete simple que se utiliza para datos de usuario.

### paquete de administración

Un paquete creado para los datos utilizados para administrar objetos de IBM MQ enviando mensajes de administración a un servidor de mandatos. El paquete de administración supone automáticamente determinadas opciones tal como se describe en [“Creación y supresión de paquetes de datos”](#) en la página 66.

### paquete de mandatos

También se crea un paquete para mandatos para administrar objetos de IBM MQ. Sin embargo, a diferencia del paquete de administración, el paquete de mandatos no implica automáticamente determinadas opciones aunque estas opciones están disponibles. Para obtener más información sobre las opciones, consulte [“Creación y supresión de paquetes de datos”](#) en la página 66.

### paquete de grupo

Un paquete que se utiliza para mantener un conjunto de elementos de datos agrupados. Los paquetes de grupo no pueden utilizarse para administrar objetos de IBM MQ.

Además, el **paquete de sistema** lo crea la MQAI cuando se devuelve un mensaje de respuesta desde el servidor de mandatos y se coloca en el paquete de salida de un usuario. Un paquete de sistema no puede ser modificado por el usuario.

Utilización de paquetes de datos. En este tema se listan las distintas formas de utilizar paquetes de datos:

## Utilización de paquetes de datos

En la siguiente lista se muestran las distintas formas de utilizar paquetes de datos:

- Puede crear y suprimir paquetes de datos [“Creación y supresión de paquetes de datos”](#) en la página 66.
- Puede enviar datos entre aplicaciones utilizando paquetes de datos [“Colocación y recepción de paquetes de datos mediante la MQAI”](#) en la página 67.
- Puede añadir elementos de datos a paquetes de datos [“Adición de elementos de datos a paquetes con la MQAI”](#) en la página 68.
- Puede añadir un mandato de consulta dentro de un paquete de datos de [“Adición de un mandato de consulta a un paquete”](#) en la página 69.
- Puede consultar dentro de paquetes de datos [“Realizar consultas dentro de paquetes de datos”](#) en la página 69.
- Puede contar los elementos de datos dentro de un paquete de datos [“Recuento de elementos de datos”](#) en la página 72.
- Puede cambiar información dentro de un paquete de datos [“Cambiar información dentro de un paquete”](#) en la página 70.
- Puede borrar un paquete de datos [“Borrado de un paquete utilizando la llamada mqClearBag”](#) en la página 71.
- Puede truncar un paquete de datos [“Truncamiento de un paquete utilizando la llamada mqTruncateBag”](#) en la página 71.
- Puede convertir paquetes y almacenamientos intermedios [“Conversión de paquetes y almacenamientos intermedios”](#) en la página 71.

### Creación y supresión de paquetes de datos

#### Creación de paquetes de datos

Para utilizar la MQAI, primero debe crear un paquete de datos utilizando la llamada mqCreateBag. Como entrada para esta llamada, debe proporcionar una o más opciones para controlar la creación del paquete.

El parámetro **Options** de la llamada MQCreateBag le permite elegir si desea crear un paquete de usuario, un paquete de mandatos, un paquete de grupo o un paquete de administración.

Para crear un paquete de usuario, un paquete de mandatos, o un paquete de grupo, puede elegir una o más opciones adicionales para:

- Utilizar el formato de lista cuando haya dos o más apariciones adyacentes del mismo selector de un paquete.
- Volver a ordenar los elementos de datos cuando se añaden a un mensaje PCF para asegurarse de que los parámetros están en el orden correcto. Para obtener más información sobre los elementos de datos, consulte [“Tipos de elemento de datos disponibles en la MQAI”](#) en la página 67.
- Comprobar los valores de selectores de usuario para los elementos que se añaden al paquete.

Los paquetes de administración implican automáticamente estas opciones.

Un paquete de datos se identifica por su manejador. El manejador de paquete se devuelve de mqCreateBag y debe suministrarse en todas las otras llamadas que utilizan el paquete de datos.

Para obtener una descripción completa de la llamada mqCreateBag, consulte [mqCreateBag](#).

#### Supresión de paquetes de datos

Cualquier paquete de datos creado por el usuario también debe suprimirse utilizando la llamada `mqDeleteBag`. Por ejemplo, si se crea un paquete en el código de usuario, también debe suprimirse en el código de usuario.

La MQAI crea y suprime automáticamente paquetes de sistema. Para obtener más información, consulte el apartado [“Envío de mandatos de administración al servidor de mandatos qm utilizando la llamada `mqExecute`”](#) en la página 73. El código de usuario no puede suprimir un paquete de sistema.

Para obtener una descripción completa de la llamada `mqDeleteBag`, consulte [mqDeleteBag](#).

Multi

### **Colocación y recepción de paquetes de datos mediante la MQAI**

Los datos también se pueden enviar entre aplicaciones transfiriendo y obteniendo paquetes de datos utilizando las llamadas `mqPutBag` y `mqGetBag`. Esto permite a la Interfaz administrativa de IBM MQ (MQAI) gestionar el almacenamiento intermedio en lugar de la aplicación.

La llamada `mqPutBag` convierte el contenido del paquete especificado en un mensaje PCF y envía el mensaje a la cola especificada y la llamada `mqGetBag` elimina el mensaje de la cola especificada y la vuelve a convertir en un paquete de datos. Por lo tanto, la llamada `mqPutBag` es equivalente a la llamada `mqBagToBuffer` seguida de `MQPUT` y la llamada `mqGetBag` es equivalente a la llamada `MQGET` seguida de `mqBufferToBag`.

Para obtener más información sobre el envío y recepción de mensajes PCF en una cola específica, consulte [“Envío y recepción de mensajes PCF en una cola especificada”](#) en la página 28

**Nota:** Si decide utilizar la llamada `mqGetBag`, los detalles de PCF dentro del mensaje deben ser correctos; si no, se produce el correspondiente error y no se devuelve el mensaje PCF.

Multi

### **Tipos de elemento de datos disponibles en la MQAI**

La Interfaz administrativa de IBM MQ (MQAI) utiliza elementos de datos para llenar los paquetes de datos cuando estos se crean. Estos elementos de datos pueden ser elementos del usuario o del sistema.

Estos elementos del usuario contienen datos de usuario como atributos de objetos que se están administrando. Los elementos del sistema deben utilizarse para un tener mayor control sobre los mensajes generados: por ejemplo, la generación de las cabeceras de mensaje. Para obtener más información sobre elementos del sistema, consulte [“Elementos del sistema y MQAI”](#) en la página 68.

## **Tipos de elementos de datos**

Cuando haya creado un paquete de datos, puede rellenarlo con elementos de enteros o elementos de series de caracteres. Puede consultar acerca de los tres tipos de elementos.

El elemento de datos puede ser un elemento de entero o un elemento de serie de caracteres. A continuación se muestran los tipos de elementos de datos disponibles dentro de la MQAI:

- Entero
- Entero de 64 bits
- Filtro de entero
- Serie de caracteres
- Filtro de texto
- Serie de bytes
- Filtro de serie de bytes
- Manejador de paquete

## **Utilización de elementos de datos**

A continuación se muestran las diversas formas de utilización de elementos de datos:

- [“Recuento de elementos de datos”](#) en la página 72.
- [“Supresión de elementos de datos”](#) en la página 72.

- “Adición de elementos de datos a paquetes con la MQAI” en la página 68.
- “Filtrado y consulta de elementos de datos” en la página 69.

**Multi** Elementos del sistema y MQAI

La Interfaz de administración de IBM MQ (MQAI) puede utilizar elementos del sistema para:

- La generación de cabeceras PCF. Los elementos del sistema puede controlar el identificador de mandatos PCF, las opciones de control, el número de secuencia de mensaje y el tipo de mandato.
- Conversión de datos. Los elementos del sistema manejan el identificador de juego de caracteres para los elementos de serie de caracteres del paquete.

Al igual que todos los elementos de datos, los elementos del sistema constan de un selector y un valor. Para obtener información sobre estos selectores y para lo que sirven, consulte [Selectores MQAI](#).

Los elementos del sistema son exclusivos. Uno o varios elementos del sistema se pueden identificar mediante un selector del sistema. Sólo hay una aparición de cada selector del sistema.

La mayoría de los elementos del sistema pueden modificarse (consulte “Cambiar información dentro de un paquete” en la página 70), pero el usuario no puede modificar las opciones de creación de paquete. No puede suprimir elementos del sistema. (Consulte el apartado “Supresión de elementos de datos” en la página 72).

**Multi** Adición de elementos de datos a paquetes con la MQAI

Cuando se crea un paquete de datos mediante la Interfaz administrativa de IBM MQ (MQAI), puede rellenarlo con elementos de datos. Estos elementos de datos pueden ser elementos del usuario o del sistema.

Para obtener más información sobre los elementos de datos, consulte “Tipos de elemento de datos disponibles en la MQAI” en la página 67.

La MQAI permite añadir elementos de entero, elementos de entero de 64 bits, elementos de filtro de enteros, elementos de serie de caracteres, un filtro de texto, elementos de serie de bytes y elementos de filtro de serie de bytes a paquetes y esto se muestra en la [Figura 5 en la página 68](#). Los elementos se identifican mediante un selector. Normalmente un selector identifica un solo elemento, pero este no siempre es el caso. Si un elemento de datos con el selector especificado ya existe en el paquete, se añade una instancia adicional de ese selector al final del paquete.

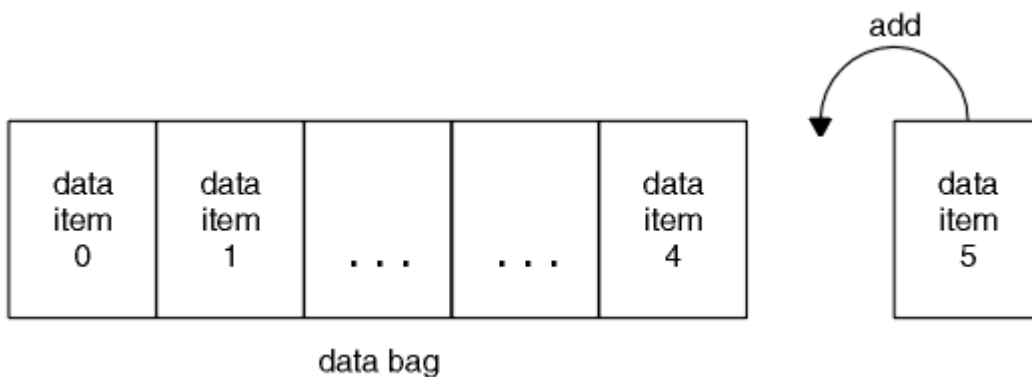


Figura 5. Adición de elementos de datos

Añadir elementos de datos a un paquete utilizando las llamadas mqAdd\*:

- Para añadir elementos de entero, utilice la llamada mqAddInteger tal como se describe en [mqAddInteger](#)
- Para añadir elementos de entero de 64 bits, utilice la llamada mqAddInteger64 tal como se describe en [mqAddInteger64](#)
- Para añadir elementos de filtro de enteros, utilice la llamada mqAddIntegerFilter tal como se describe en [mqAddIntegerFilter](#)

- Para añadir elementos de serie de caracteres, utilice la llamada `mqAddString` tal como se describe en [mqAddString](#)
- Para añadir elementos de filtro de texto, utilice la llamada `mqAddStringFilter` tal como se describe en [mqAddStringFilter](#)
- Para añadir elementos de serie de bytes, utilice la llamada `mqAddByteString` tal como se describe en [mqAddByteString](#)
- Para añadir elementos de filtro de serie de bytes, utilice la llamada `mqAddByteStringFilter` tal como se describe en [mqAddByteStringFilter](#)

Para obtener más información sobre la adición de elementos de datos a un paquete, consulte [“Elementos del sistema y MQAI”](#) en la página 68

#### Multi

#### Adición de un mandato de consulta a un paquete

La llamada `mqAddInquiry` se utiliza para añadir un mandato de consulta a un paquete. La llamada es específicamente para fines de administración, por lo que sólo puede utilizarse con paquetes de administración. Le permite especificar los selectores de los atributos que desea consultar desde IBM MQ. Para obtener una descripción completa de la llamada `mqAddInquiry`, consulte [mqAddInquiry](#).

#### Multi

#### Filtrado y consulta de elementos de datos

Cuando se utiliza la MQAI para consultar sobre los atributos de objetos de IBM MQ, puede controlar los datos que se devuelven al programa de dos formas.

- Puede **filtrar** los datos que se devuelven utilizando las llamadas `mqAddInteger` y `mqAddString`. Este enfoque permite especificar un par de *Selector* y *ItemValue*, por ejemplo:

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

Este ejemplo especifica que el tipo de cola (*Selector*) debe ser local (*ItemValue*) y esta especificación debe coincidir con los atributos del objeto (en este caso, una cola) sobre el que se realiza la consulta.

Otros atributos que se pueden filtrar corresponden a los mandatos *Inquire\** de PCF que se pueden encontrar en [“Introducción a los Formatos de mandato programable de IBM MQ”](#) en la página 25. Por ejemplo, para consultar sobre los atributos de un canal, consulte el mandato *Consultar canal* en esta documentación del producto. Los "parámetros obligatorios" y los "parámetros opcionales" del mandato *Consultar canal* identifican los selectores que puede utilizar para el filtrado.

- Puede **consultar** atributos concretos de un objeto utilizando la llamada `mqAddInquiry`. Esto especifica el selector en los que está interesado. Si no especifica el selector, se devuelven todos los atributos del objeto.

A continuación se muestra un ejemplo del filtro y la consulta de los atributos de una cola:

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

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

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

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

#### Multi

#### Realizar consultas dentro de paquetes de datos

Puede realizar consultas acerca de:

- El valor de un elemento de entero utilizando la llamada mqInquireInteger. Consulte [mqInquireInteger](#).
- El valor de un elemento de entero de 64 bits utilizando la llamada mqInquireInteger64. Consulte [mqInquireInteger64](#).
- El valor de un elemento de filtro de enteros utilizando la llamada mqInquireIntegerFilter. Consulte [mqInquireIntegerFilter](#).
- El valor de un elemento de serie de caracteres utilizando la llamada mqInquireString. Consulte [mqInquireString](#).
- El valor de un elemento de filtro de texto utilizando la llamada mqInquireStringFilter. Consulte [mqInquireStringFilter](#).
- El valor de un elemento de serie de bytes utilizando la llamada mqInquireByteString. Consulte [mqInquireByteString](#).
- El valor de un elemento de filtro de serie de bytes utilizando la llamada mqInquireByteStringFilter. Consulte [mqInquireByteStringFilter](#).
- El valor de un manejador de paquete utilizando la llamada mqInquireBag. Consulte [mqInquireBag](#).

También puede consultar sobre el tipo (entero, entero de 64 bits, filtro de enteros, serie de caracteres, filtro de texto, serie de bytes, filtro de serie de bytes o manejador de paquete) de un elemento específico utilizando la llamada mqInquireItemInfo. Consulte [mqInquireItemInfo](#).

**Multi** Cambiar información dentro de un paquete

La MQAI permite cambiar información dentro de un paquete utilizando las llamadas mqSet\*. Puede:

1. Modificar elementos de datos dentro de un paquete. El índice permite sustituir una instancia individual de un parámetro identificando la aparición del elemento que se debe modificar (consulte [Figura 6](#) en la página 70).

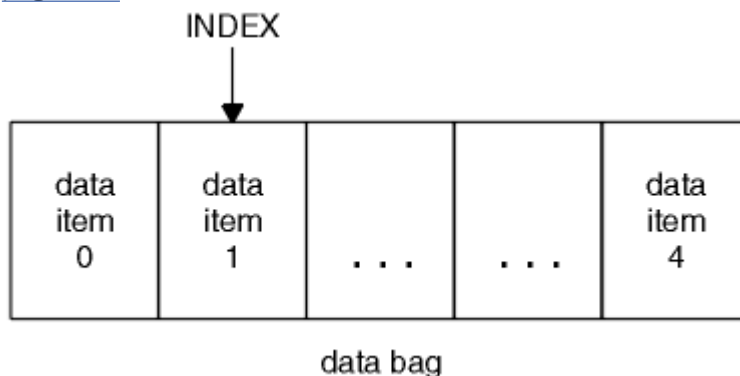


Figura 6. Modificación de un único elemento de datos

2. Suprimir todas las apariciones existentes del selector especificado y añadir una nueva aparición al final del paquete. (Consulte el apartado [Figura 7](#) en la página 70). Un valor de índice especial permite sustituir **todas** las instancias de un parámetro.

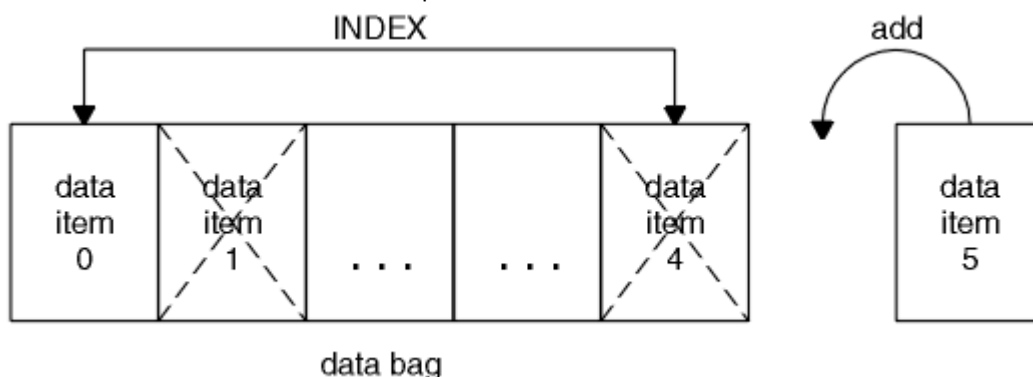


Figura 7. Modificación de todos los elementos de datos

**Nota:** El índice conserva el orden de inserción en el paquete aunque puede afectar a los índices de otros elementos de datos.

La llamada `mqSetInteger` permite modificar elementos de entero dentro de un paquete. La llamada `mqSetInteger64` permite modificar elementos de enteros de 64 bits. La llamada `mqSetIntegerFilter` permite modificar elementos de filtro de enteros. La llamada `mqSetString` permite modificar elementos de serie de caracteres. La llamada `mqSetStringFilter` permite modificar elementos de filtro de texto. La llamada `mqSetByteString` permite modificar elementos de serie de bytes. La llamada `mqSetByteStringFilter` permite modificar elementos de filtro de serie de bytes. Como alternativa, puede utilizar estas llamadas para suprimir todas las apariciones existentes del selector especificado y añadir una nueva aparición al final del paquete. El elemento de datos puede ser un elemento de usuario o un elemento de sistema.

Para obtener una descripción completa de estas llamadas, consulte:

- [mqSetInteger](#)
- [mqSetInteger64](#)
- [mqSetIntegerFilter](#)
- [mqSetString](#)
- [mqSetStringFilter](#)
- [mqSetByteString](#)
- [mqSetByteStringFilter](#)

#### **Multi** Borrado de un paquete utilizando la llamada `mqClearBag`

La llamada `mqClearBag` elimina todos los elementos de usuario de un paquete de usuario y restablece los elementos del sistema a sus valores iniciales. También se suprimen los paquetes de sistema incluidos dentro del paquete.

Para obtener una descripción completa de la llamada `mqClearBag`, consulte [mqClearBag](#).

#### **Multi** Truncamiento de un paquete utilizando la llamada `mqTruncateBag`

La llamada `mqTruncateBag` reduce el número de elementos de usuario de un paquete de usuario suprimiendo los elementos del final del paquete, empezando por el elemento añadido más recientemente. Por ejemplo, puede usarse cuando se utiliza la misma información de cabecera para generar más de un mensaje.

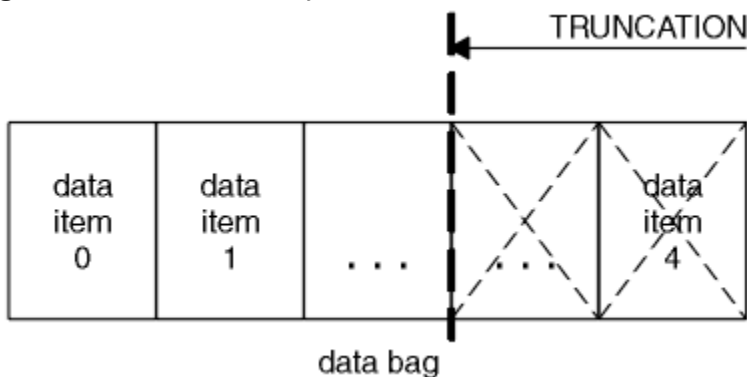


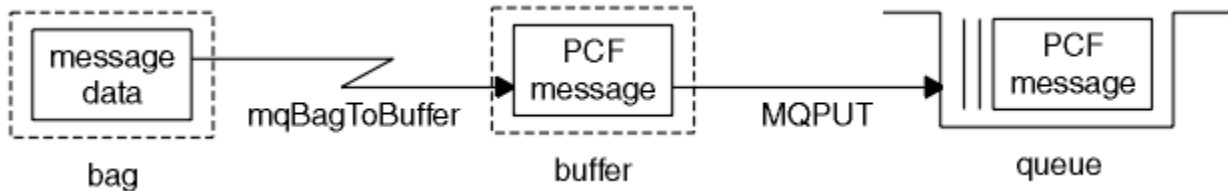
Figura 8. Truncar un paquete

Para obtener una descripción completa de la llamada `mqTruncateBag`, consulte [mqTruncateBag](#).

#### **Multi** Conversión de paquetes y almacenamientos intermedios

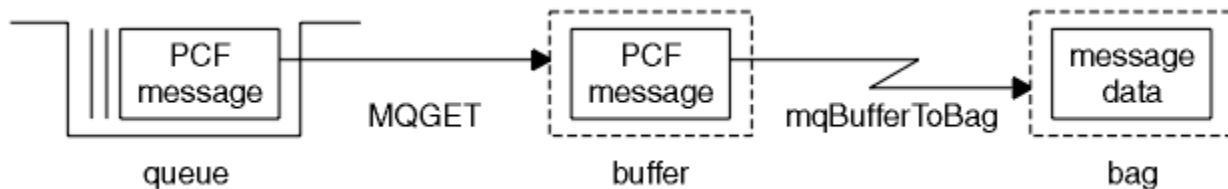
Para enviar datos entre aplicaciones, primero los datos del mensaje se colocan en un paquete. A continuación, los datos del paquete se convierten en un mensaje PCF utilizando la llamada `mqBagToBuffer`. El mensaje PCF se envía a la cola necesaria utilizando la llamada `MQPOT`. Esto se

muestra en la [Figura 9](#) en la página 72. Para obtener una descripción completa de la llamada `mqBagToBuffer`, consulte [mqBagToBuffer](#).



*Figura 9. Conversión de paquetes en mensajes PCF*

Para recibir datos, el mensaje se recibe en un almacenamiento intermedio utilizando la llamada `MQGET`. Los datos del almacenamiento intermedio a continuación se convierten en un paquete utilizando la llamada `mqBufferToBag`, siempre que el almacenamiento intermedio contenga un mensaje PCF válido. Esto se muestra en la [Figura 10](#) en la página 72. Para obtener una descripción completa de la llamada `mqBufferToBag`, consulte [mqBufferToBag](#).



*Figura 10. Conversión de mensajes PCF a formato de paquete*

#### Multi Recuento de elementos de datos

La llamada `mqCountItems` cuenta el número de elementos de usuario, elementos del sistema, o ambos, que están almacenados en un paquete de datos, y devuelve este número. Por ejemplo, `mqCountItems (Bolsa, 7, ...)`, devuelve el número de elementos de la bolsa con un selector de 7. Puede contar elementos por selector individual, por selectores de usuario, por selectores del sistema o por todos los selectores.

**Nota:** Esta llamada cuenta el número de elementos de datos, no el número de selectores exclusivos del paquete. Un selector puede darse varias veces, por lo que puede haber menos selectores exclusivos en el paquete que elementos de datos.

Para obtener una descripción completa de la llamada `mqCountItems`, consulte [mqCountItems](#).

#### Multi Supresión de elementos de datos

Puede suprimir elementos de paquetes de varias formas. Puede:

- Eliminar uno o más elementos de usuario de un paquete. Para obtener información detallada, consulte [“Supresión de elementos de datos de un paquete utilizando la llamada mqDeleteItem”](#) en la página 72.
- Suprimir todos los elementos de usuario de un paquete, es decir, borrar un paquete. Para obtener información detallada, consulte [“Borrado de un paquete utilizando la llamada mqClearBag”](#) en la página 71.
- Suprimir todos los elementos de usuario del final de un paquete, es decir, truncar un paquete. Para obtener información detallada, consulte [“Truncamiento de un paquete utilizando la llamada mqTruncateBag”](#) en la página 71.

#### Multi Supresión de elementos de datos de un paquete utilizando la llamada mqDeleteItem

La llamada `mqDeleteItem` elimina uno o más elementos de usuario de un paquete. El índice se utiliza para suprimir uno de los siguientes valores:

1. Una única aparición del selector especificado. (Consulte [Figura 11](#) en la página 73.)



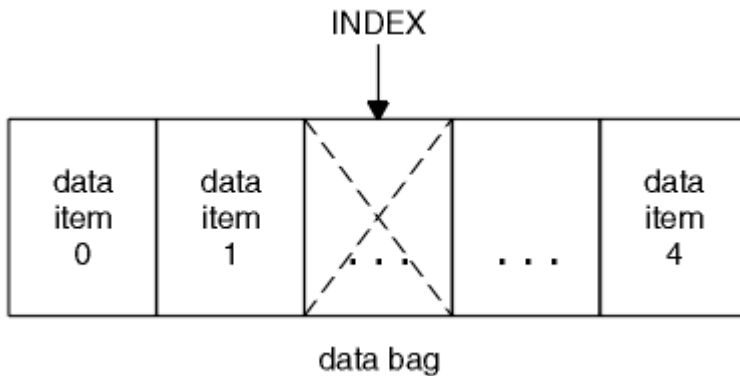


Figura 11. Supresión de un único elemento de datos

o

2. Todas las apariciones del selector especificado. (Consulte [Figura 12 en la página 73.](#))

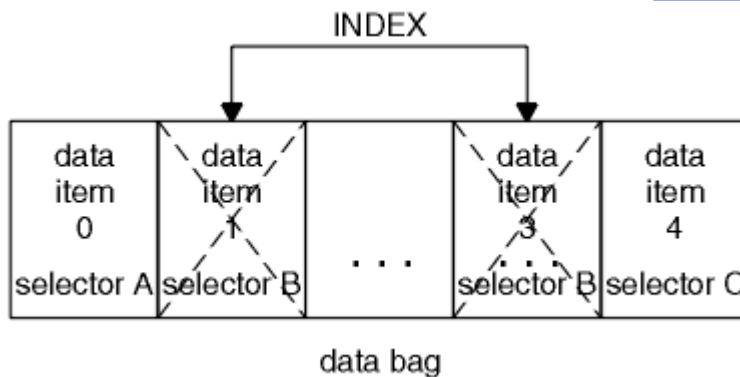


Figura 12. Supresión de todos los elementos de datos

**Nota:** El índice conserva el orden de inserción en el paquete aunque puede afectar a los índices de otros elementos de datos. Por ejemplo, la llamada `mqDeleteItem` no conserva los valores de índice de los elementos de datos que siguen el elemento suprimido porque los índices se reorganizan para rellenar el hueco que permanece del elemento suprimido.

Para obtener una descripción completa de la llamada `mqDeleteItem`, consulte [mqDeleteItem](#).

## **Multi Envío de mandatos de administración al servidor de mandatos qm utilizando la llamada mqExecute**

Una vez que se ha creado y llenado un paquete de datos, puede enviarse un mensaje de mandato administrativo al servidor de mandatos de un gestor de colas utilizando la llamada `mqExecute`. Esto maneja el intercambio con el servidor de mandatos y devuelve respuestas en un paquete.

Después de haber creado y llenado los paquetes de datos, puede enviar un mensaje de mandato de administración al servidor de mandatos de un gestor de colas. La forma más fácil de hacerlo es utilizando la llamada `mqExecute`. La llamada `mqExecute` envía un mensaje de mandato de administración como mensaje no persistente y espera las respuestas. Las respuestas se devuelven en un paquete de respuesta. Por ejemplo, estas pueden incluir información sobre los atributos relacionados con varios objetos de IBM MQ o una serie de mensajes de respuesta de error PCF. Por lo tanto, el paquete de respuesta podría contener un código de retorno o sólo *paquetes anidados*.

Los mensajes de respuesta se colocan en paquetes del sistema que crea el sistema. Por ejemplo, para consultas sobre los nombres de los objetos, se crea un paquete de sistema para mantener esos nombres de objeto y el paquete se inserta en el paquete de usuario. Los manejadores de estos paquetes se insertarán en el paquete de respuesta y el selector `MQHA_BAG_HANDLE` podrá acceder al paquete anidado. El paquete de sistema permanece en el almacenamiento, si no se suprime, hasta que se suprime el paquete de respuestas.

El concepto de *anidamiento* es muestra en la [Figura 13](#) en la página 74.

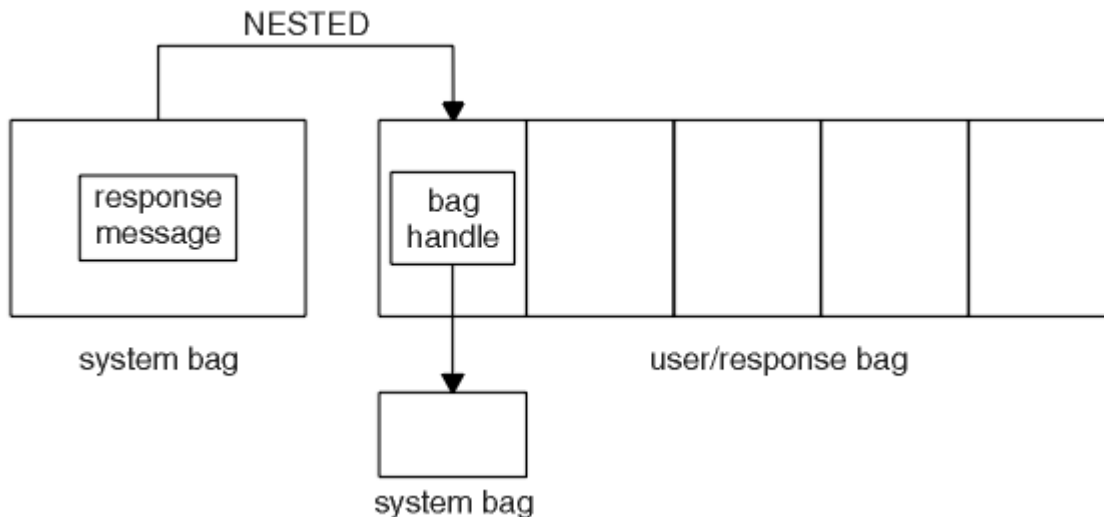


Figura 13. Anidamiento

Como entrada para la llamada `mqExecute` debe proporcionar:

- Un manejador de conexión MQI.
- El mandato que se va a ejecutar. Este debe ser uno de los valores `MQCMD_*`.

**Nota:** Si la MQAI no reconoce este valor, el valor se sigue aceptando. Sin embargo, si se utilizó la llamada `mqAddInquiry` para insertar valores en el paquete, este parámetro debe ser un mandato `INQUIRE` reconocido por la MQAI. Es decir, el parámetro debe tener el formato `MQCMD_INQUIRE_*`.

- Opcionalmente, un manejador del paquete que contiene las opciones que controlan el proceso de la llamada. Aquí también es donde puede especificar el tiempo máximo en milisegundos que la MQAI debe esperar para cada mensaje de respuesta.
- Un manejador del paquete de administración que contiene detalles del mandato de administración que va a emitirse.
- Un manejador del mensaje de respuesta que recibe los mensajes de respuesta.

Los siguientes manejadores son opcionales:

- Un manejador de objeto de la cola donde se colocará el mandato de administración.

Si no se especifica ningún manejador de objeto, el mandato de administración se coloca en la cola `SYSTEM.ADMIN.COMMAND.QUEUE` que pertenece al gestor de colas conectado actualmente. Éste es el valor predeterminado.

- Un manejador de objeto de la cola donde se colocarán los mensajes de respuesta.

Puede elegir colocar los mensajes de respuesta en una cola dinámica que la MQAI crea automáticamente. La cola creada sólo existe durante la llamada y la MQAI la suprime al salir de la llamada `mqExecute`.

Para obtener ejemplos de usos de la llamada `mqExecute`, consulte el apartado [Código de ejemplo](#)

## V 9.1.0 Administración utilizando REST API

Puede utilizar la administrative REST API para administrar objetos IBM MQ como, por ejemplo, gestores de colas y colas, y agentes y transferencias Managed File Transfer. La información se envía a y se recibe de administrative REST API en formato JSON. Estas API RESTful pueden ayudarle a incorporar la administración de IBM MQ en las herramientas populares de automatización y DevOps.

## Antes de empezar

Para ver la información de referencia sobre los recursos REST disponibles, consulte [Referencia de la administrative REST API](#).

## Procedimiento

- [“Iniciación a administrative REST API” en la página 75](#)
- [“Utilización de administrative REST API” en la página 78](#)
- [“Administración remota mediante REST API” en la página 80](#)
- [“Indicaciones de fecha y hora de REST API” en la página 84](#)
- [“Manejo de errores de REST API” en la página 84](#)
- [“Descubrimiento de REST API” en la página 87](#)
- [“Soporte multilingüístico de REST API” en la página 88](#)



## V 9.1.0 Iniciación a administrative REST API


Empiece rápidamente con la administrative REST API y pruebe unas pocas solicitudes de ejemplo utilizando cURL para crear, actualizar, ver y suprimir una cola.

## Antes de empezar

Para empezar a utilizar la administrative REST API, los ejemplos de esta tarea tienen los requisitos siguientes:

- Los ejemplos utilizan cURL para realizar solicitudes REST para mostrar información sobre gestores de colas en el sistema, y para crear una cola, actualizar, ver y suprimir una cola. Por lo tanto, para completar esta tarea, necesita tener el cURL instalado en el sistema.
- Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar el mandato **dspmweb**:


-  **z/OS** En z/OS, debe tener autorización para ejecutar el mandato **dspmweb** y acceso de escritura en el archivo `mqwebuser.xml`.
-  **Multi** En todos los demás sistemas operativos, debe ser un [usuario con privilegios](#).

 **IBM i** En IBM i, los mandatos deben estar en ejecución en QSHELL.

## Procedimiento

1. Si el servidor mqweb todavía no está configurado para que lo utilice administrative REST API, administrative REST API para MFT, messaging REST API o IBM MQ Console, configure el servidor mqweb.

Para obtener más información sobre la creación de una configuración básica para el servidor mqweb con un registro básico, consulte [Configuración básica para el servidor mqweb](#).

2.  **z/OS**  
En z/OS, establezca la variable de entorno `WLP_USER_DIR` para que pueda utilizar el mandato **dspmweb**. Establezca la variable para que apunte a la configuración del servidor mqweb especificando el mandato siguiente.

```
export WLP_USER_DIR=WLP_user_directory
```

donde `WLP_user_directory` es el nombre del directorio que se pasa a `crtmqweb`. Por ejemplo:

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

Si desea más información, consulte [Creación del servidor mqweb](#).

3. Determine el URL de REST API URL especificando el mandato siguiente:

```
dspmqweb status
```

Los ejemplos de los pasos siguientes presuponen que el URL de REST API URL es el URL de URL `https://localhost:9443/ibmmq/rest/v1/`. Si el URL es diferente al valor predeterminado, sustituya el URL en los pasos siguientes.

4. Pruebe una solicitud GET en el recurso `qmgr` utilizando autenticación básica con el usuario `mqadmin`:

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr -X GET -u mqadmin:mqadmin
```

5. Cree, vea, modifique y suprima una cola utilizando el recurso `mqsc`:

Este ejemplo utiliza un gestor de colas QM1. Cree un gestor de colas con el mismo nombre o sustituya un gestor de colas existentes en el sistema.

a) Realice una solicitud POST en el recurso `mqsc` para crear la cola local:

En el cuerpo de la solicitud, el nombre de la nueva cola se establece en `Q1`. Se utiliza la autenticación básica y una cabecera HTTP `ibm-mq-rest-csrf-token` con un valor arbitrario se establece en la solicitud REST cURL. Esta cabecera adicional es necesaria para solicitudes POST, PATCH y DELETE:

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/action/qmgr/QM1/mqsc -X POST -u mqadmin:mqadmin -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json" --data '{"type": "runCommandJSON", "command": "define", "qualifier": "qlocal", "name": "Q1"}'
```

b) Realice una solicitud POST en el recurso `mqsc` para ver la cola local creada en el paso “5.a” en la [página 76](#):

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/action/qmgr/QM1/mqsc -X POST -u mqadmin:mqadmin -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json" --data '{"type": "runCommandJSON", "command": "display", "qualifier": "qlocal", "name": "Q1"}'
```

c) Realice una solicitud POST en el recurso `mqsc` al recurso para actualizar la descripción de la cola:

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/action/qmgr/QM1/mqsc -X POST -u mqadmin:mqadmin -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json" --data '{"type": "runCommandJSON", "command": "alter", "qualifier": "qlocal", "name": "Q1", "parameters": {"descr": "new description"}'}
```

d) Realice una solicitud POST en el recurso `mqsc` para ver la nueva descripción de cola. Especifique el atributo **responseParameters** en el cuerpo de la solicitud de modo que la respuesta incluya el campo de descripción:

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/action/qmgr/QM1/mqsc -X POST -u mqadmin:mqadmin -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json" --data '{"type": "runCommandJSON", "command": "display", "qualifier": "qlocal", "name": "Q1", "responseParameters": [{"descr"}]}'
```

e) Realice una solicitud POST en el recurso `mqsc` para suprimir la cola:

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/action/qmgr/QM1/mqsc -X POST -u mqadmin:mqadmin -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json" --data '{"type": "runCommandJSON", "command": "delete", "qualifier": "qlocal", "name": "Q1"}'
```

f) Realice una solicitud POST en el recurso `mqsc` para probar que la cola se ha suprimido:

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/action/qmgr/QM1/mqsc -X POST -u mqadmin:mqadmin -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json" --data '{"type": "runCommandJSON", "command": "display", "qualifier": "qlocal", "name": "Q1"}'
```



## Qué hacer a continuación

- Los ejemplos utilizan la autenticación básica para proteger la solicitud. En su lugar, puede utilizar la autenticación basada en señal o la autenticación basada en cliente. Si desea más información, consulte [Utilización de autenticación de certificado de cliente con la REST API y IBM MQ Console](#), y [Utilización de la autenticación basada en señal con la REST API](#).
- Obtenga más información sobre cómo utilizar administrative REST API y construir los URL con parámetros de consulta: [“Utilización de administrative REST API”](#) en la página 78.
- Examine la información de referencia para los recursos de administrative REST API disponibles y todos los parámetros de consulta opcionales disponibles: [Referencia de administrative REST API](#).
- Obtenga información sobre cómo utilizar la administrative REST API para administrar objetos IBM MQ en sistemas remotos: [“Administración remota mediante REST API”](#) en la página 80.
- Obtenga más información sobre cómo utilizar la administrative REST API con MFT: [“Cómo empezar con REST API para MFT”](#) en la página 77.
- Descubra messaging REST API, una interfaz RESTful para la mensajería IBM MQ: [Mensajería utilizando REST API](#).
- Descubra IBM MQ Console, una GUI basada en navegador: [“Administración utilizando una consola web”](#) en la página 91.


## Cómo empezar con REST API para MFT

Empiece rápidamente con administrative REST API para Managed File Transfer y pruebe algunas solicitudes de ejemplo para ver el estado del agente MFT y para ver una lista de transferencias.

### Antes de empezar

- Los ejemplos utilizan cURL para enviar solicitudes REST para ver una lista de transferencias y ver el estado del agente MFT. Por lo tanto, para completar estar, necesita tener el cURL instalado en el sistema.
- Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar el mandato **dspmqweb**:
  -  En z/OS, debe tener autorización para ejecutar el mandato **dspmqweb** y acceso de escritura en el archivo `mqwebuser.xml`.
  -  En todos los demás sistemas operativos, debe ser un [usuario con privilegios](#).

### Procedimiento

1. Asegúrese de que el servidor mqweb se ha configurado para administrative REST API para MFT:
  - Si el servidor mqweb todavía no está configurado para que lo utilice administrative REST API, administrative REST API para MFT, messaging REST API o IBM MQ Console, configure el servidor mqweb. Para obtener más información sobre la creación de una configuración básica para el servidor mqweb con un registro básico, consulte [Configuración básica para el servidor mqweb](#).
  - Si el servidor mqweb está configurado, asegúrese de que el paso 8 de la [Configuración básica para el servidor mqweb](#) se ha completado para habilitar administrative REST API para MFT.
2.  En z/OS, establezca la variable de entorno WLP\_USER\_DIR para que pueda utilizar el mandato **dspmqweb**. Establezca la variable para que apunte a la configuración del servidor mqweb especificando el mandato siguiente.

```
export WLP_USER_DIR=WLP_user_directory
```

donde *WLP\_user\_directory* es el nombre del directorio que se pasa a *crtmqweb*. Por ejemplo:

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

Si desea más información, consulte [Creación del servidor mqweb](#).

3. Determine el URL de REST API URL especificando el mandato siguiente:

```
dspmqweb status
```

Los ejemplos de los pasos siguientes presuponen que el URL de REST API URL es el URL de URL `https://localhost:9443/ibmmq/rest/v1/`. Si el URL es diferente al valor predeterminado, sustituya el URL en los pasos siguientes.

4. Realice una solicitud GET en el recurso `agent` para volver a detalles básicos sobre todos los agentes, incluyendo el nombre, tipo y estado:

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/mft/agent/ -X GET -u mftadmin:mftadmin
```

5. Cree algunas transferencias para mostrar utilizando el mandato **fteCreateTransfer**.

El servidor *mqweb* almacena en la memoria caché información sobre transferencias y devuelve esta información cuando se realiza una solicitud. Esta memoria caché se restablece cuando se reinicia el servidor *mqweb*. Puede ver si el servidor se ha reiniciado visualizando los archivos `console.log` y `messages.log` o en z/OS, mirando la salida de la tarea iniciada.

6. Realice una solicitud GET en el recurso `transfer` para devolver detalles de hasta cuatro transferencias que se han realizado desde que se inició el servidor *mqweb*:

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/mft/transfer?limit=4 -X GET -u mftadmin:mftadmin
```

## Qué hacer a continuación

- Los ejemplos utilizan la autenticación básica para proteger la solicitud. En su lugar, puede utilizar la autenticación basada en señal o la autenticación basada en cliente. Si desea más información, consulte [Utilización de la autenticación basada en señal con el REST API](#) y [Utilización de la autenticación de certificado de cliente con REST API y IBM MQ Console](#).
- Obtenga más información sobre cómo utilizar `administrative REST API` y construir los URL con parámetros de consulta: [“Utilización de administrative REST API” en la página 78](#).
- Examine la información de referencia para los recursos de `administrative REST API` para MFT disponibles y todos los parámetros de consulta opcionales disponibles: [Referencia de administrative REST API](#).
- Descubra `messaging REST API`, una interfaz RESTful para la mensajería IBM MQ: [Mensajería utilizando REST API](#).
- Descubra IBM MQ Console, una GUI basada en navegador: [“Administración utilizando una consola web” en la página 91](#).

## **V 9.1.0** Utilización de administrative REST API

Cuando utiliza `administrative REST API`, se invocan métodos HTTP en los URL que representan los diversos objetos de IBM MQ, como gestores de colas o colas. El método HTTP, por ejemplo, POST representa el tipo de acción que debe realizarse en el objeto que está representado por el URL. Es posible que se proporcione más información acerca de la acción en JSON como parte de la carga útil del método HTTP o codificada en los parámetros de consulta. La información sobre el resultado de realizar la acción puede devolverse como el cuerpo de la respuesta HTTP.

## Antes de empezar

Tenga en cuenta estos puntos antes de utilizar `administrative REST API`:

- Debe autenticarse con el servidor mqweb para poder utilizar administrative REST API. Puede autenticarse mediante la autenticación básica HTTP, la autenticación de certificados de cliente o la autenticación basada en señal. Para obtener más información sobre cómo utilizar estos métodos de autenticación, consulte [Seguridad de IBM MQ Console y REST API](#).
- REST API es sensible a las mayúsculas y minúsculas. Por ejemplo un HTTP GET sobre el URL siguiente no muestra información si el gestor de colas se llama qmgr1.

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

- No todos los caracteres que se pueden utilizar en nombres de objeto de IBM MQ se pueden codificar directamente en un URL. Para codificar estos caracteres correctamente, debe utilizar la codificación de URL apropiada:
  - Una barra inclinada, /, debe codificarse como %2F.
  - Un signo de porcentaje, %, debe codificarse como %25.
- Debido al comportamiento de algunos navegadores, no nombre objetos utilizando solo un punto o una barra inclinada.

## Acerca de esta tarea

Cuando se utiliza REST API para realizar una acción en un objeto, primero es necesario construir un URL para representar ese objeto. Cada URL empieza con un prefijo, que describe a qué nombre de host y puerto se debe enviar la solicitud. El resto del URL describe un objeto particular o un conjunto de objetos conocido como un recurso.

La acción que se debe realizar en el recurso define si el URL necesita parámetros de consulta o no. También define el método HTTP que se utiliza y si se envía información adicional al URL, o se devuelve del mismo, en formato JSON. La información adicional puede formar parte de la solicitud HTTP o devolverse como parte de la respuesta HTTP.

Después de construir el URL y de crear una carga útil JSON opcional para enviar en la solicitud HTTP, puede enviar la solicitud HTTP a IBM MQ. Puede enviar la solicitud utilizando la implementación de HTTP que se base en el lenguaje de programación de su elección. También puede enviar las solicitudes utilizando herramientas de línea de mandatos como cURL, o un navegador web o complemento de navegador web.

**Importante:** Como mínimo, hay que seguir los pasos [“1.a” en la página 79](#) y [“1.b” en la página 79](#).

## Procedimiento

1. Construya el URL:

a) Determine el URL del prefijo especificando el mandato siguiente:

```
dspmweb status
```

El URL que desea utilizar incluye la frase `/ibmmq/rest/`.

b) Añada el recurso a la vía de acceso de URL.

Están disponibles los recursos IBM MQ:

- [/admin/installation](#)
- [/admin/qmgr](#)
- [/admin/queue](#)
- [/admin/subscription](#)
- [/admin/channel](#)
- [/action/qmgr/{qmgrname}/mqsc](#)

Están disponibles los recursos Managed File Transfer:

- [/admin/agent](#)
- [/admin/transfer](#)
- [/admin/monitor](#)

Por ejemplo, para interactuar con gestores de colas, añada `/qmgr` al URL de prefijo para crear el URL siguiente:

```
https://localhost:9443/ibmmq/rest/v2/admin/qmgr
```

c) Opcional: Añada los segmentos de vía de acceso opcionales adicionales al URL.

En la información de referencia para cada tipo de objeto, los segmentos opcionales pueden identificarse en el URL por las llaves que los incluyen `{ }`.

Por ejemplo, añada el nombre de gestor de colas `QM1` al URL para crear el URL siguiente:

```
https://localhost:9443/ibmmq/rest/v2/admin/qmgr/QM1
```

d) Opcional: Añada un parámetro de consulta opcional al URL.

Añada un signo de interrogación, `?`, nombre de variable, signo igual `=`, y un valor o lista de valores para el URL.

Por ejemplo, para solicitar todos los atributos del gestor de colas `QM1`, cree el URL siguiente:

```
https://localhost:9443/ibmmq/rest/v2/admin/qmgr/QM1?attributes=*
```

e) Añada parámetros de consulta opcionales adicionales al URL.

Añada un ampersand, `&`, al URL y, a continuación, repita el [paso d](#).

2. Invoque el método HTTP correspondiente en el URL. Especifique cualquier carga útil JSON opcional y proporcione las credenciales de seguridad apropiadas para la autenticación. Por ejemplo:

- Utilice la implementación de HTTP/REST del lenguaje de programación elegido.
- Utilice una herramienta como un complemento de navegador de cliente REST o cURL.

## **V 9.1.0** Administración remota mediante REST API

Puede utilizar REST API para administrar los gestores de colas remotos y los objetos de IBM MQ asociados con dichos gestores de colas. Esta administración remota incluye los gestores de colas que están en el mismo sistema, pero no en la misma instalación IBM MQ que el servidor `mqweb`. Por lo tanto, puede utilizar REST API para administrar toda la red de IBM MQ sólo con una única instalación que ejecuta el servidor `mqweb`. Para administrar los gestores de colas remotos, debe configurar la pasarela de administrative REST API de manera que al menos un gestor de colas de la misma instalación que el servidor `mqweb` actúe como un gestor de colas de pasarela. Por lo tanto, puede especificar el gestor de colas remoto en el URL de recursos de REST API para realizar la acción administrativa especificada.

### **Antes de empezar**

Puede impedir la administración remota inhabilitando la pasarela de administrative REST API. Para obtener más información, consulte [Configuración de la pasarela de administrative REST API](#).

Para utilizar la pasarela de administrative REST API, se deben cumplir las condiciones siguientes:

- El servidor `mqweb` tiene que estar configurado e iniciado. Para obtener más información sobre cómo configurar e iniciar el servidor `mqweb`, consulte [“Iniciación a administrative REST API”](#) en la página 75.
- El gestor de colas que desee configurar como gestor de colas de pasarela tiene que estar en la misma instalación que el servidor `mqweb`.
- El gestor de colas remoto que desee administrar tiene que ser IBM MQ 8.0 o posteriores.
- Hay que asegurarse de que todos los atributos especificados en la solicitud sean válidos para el sistema al que se envía dicha solicitud. Por ejemplo, si el gestor de colas de pasarela está en



Windows y el gestor de colas remoto está en z/OS, no puede solicitar que se devuelva el atributo `dataCollection.statistics` en una solicitud HTTP GET en el recurso `queue`.

- Hay que asegurarse de que todos los atributos especificados en la solicitud sean válidos para el nivel de IBM MQ al que se envía dicha solicitud. Por ejemplo, si el gestor de colas remoto está ejecutando en IBM MQ 8.0, no puede solicitar que se devuelva el atributo `extended.enableMediaImageOperations` en una solicitud HTTP GET en el recurso `queue`.
- Hay que utilizar uno de estos recursos REST soportados:
  - `/queue`
  - `/subscription`
  - `/channel`
  - `/mqsc`
  - `/qmgr`

El recurso `/qmgr` solo devuelve un subconjunto de los atributos cuando se consulta un gestor de colas remoto: `name`, `status.started`, `status.channelInitiatorState`, `status.ldapConnectionState`, `status.connectionCount` y `status.publishSubscribeState`.

## Acerca de esta tarea

Para utilizar la pasarela de administrative REST API para administrar los gestores de colas remotos, debe preparar los gestores de colas para la administración remota. Es decir, hay que configurar colas de transmisión, escuchas y canales emisor y receptor entre el gestor de colas de pasarela y el gestor de colas remoto. A continuación, se puede enviar una solicitud REST al gestor de colas remoto especificando el gestor de colas en el URL de recurso. EL gestor de colas de pasarela se especifica utilizando el mandato **setmqweb** para establecer el atributo `mqRestGatewayQmgr` en el nombre del gestor de colas de pasarela, o enviando el nombre del gestor de colas de pasarela en una cabecera que se envía con la solicitud. La solicitud se envía a través del gestor de colas de pasarela del gestor de colas remoto. La respuesta se devuelve en una cabecera que indica el gestor de colas usado como gestor de colas de pasarela.

## Procedimiento

1. Configure las comunicaciones entre el gestor de colas de pasarela y los gestores de colas remotos que desee administrar. Estos pasos de configuración son los mismos pasos que son necesarios para configurar la administración remota con `runmqsc` y PCF.  
Para obtener más información sobre estos pasos, consulte [“Configuración de gestores de colas para la administración remota”](#) en la [página 208](#).
2. Configure la seguridad en los gestores de colas remotos:
  - a) Asegúrese de que existan los correspondientes ID de usuario en el sistema en que ejecuta el gestor de colas remoto. El ID de usuario debe existir en el sistema remoto que depende del rol del usuario de REST API:
    - Si el usuario de REST API está en el grupo `MQWebAdmin` o `MQWebAdminRO`, el ID de usuario que ha iniciado el servidor `mqweb` debe existir en el sistema remoto. En IBM MQ Appliance, el usuario que inicia el servidor `mqweb` es `mqsystem`.
    - Si el usuario de REST API está en el grupo `MQWebUser`, dicho ID de usuario de REST API debe existir en el sistema remoto.
  - b) Asegúrese de que se otorguen a los ID de usuario correspondientes los niveles de autorización necesarios para acceder a los recursos de REST API adecuados en el gestor de colas remoto:
    - Autoridad para poner mensajes en `SYSTEM.ADMIN.COMMAND.QUEUE`.
    - Autoridad para poner mensajes en `SYSTEM.REST.REPLY.QUEUE`.
    - Autoridad para acceder a las colas de transmisión definidas para la administración remota.

- Autoridad para visualizar los atributos del gestor de colas.
  - Autoridad para realizar peticiones REST. Para obtener más información, consulte la sección de requisitos de seguridad de los [temas de referencia de los recursos de REST API](#).
3. Configure qué gestor de colas local se utiliza como pasarela. Se puede configurar un gestor de colas de pasarela predeterminado, especificar el gestor de colas de pasarela en una cabecera HTTP o utilizar una combinación de ambos enfoques:

- Configurar un gestor de colas de pasarela predeterminado con el mandato **setmqweb**:

```
setmqweb properties -k mqRestGatewayQmgr -v qmgrName
```

donde *qmgrName* es el nombre del gestor de colas de pasarela.

Este gestor de colas de pasarela se utiliza cuando se cumplen las dos sentencias siguientes:

- No se ha especificado un gestor de colas en la cabecera `ibm-mq-rest-gateway-qmgr` de una solicitud REST.
  - El gestor de colas especificado en el URL del recurso de REST API no es un gestor de colas local.
- Configure el gestor de colas de pasarela en cada solicitud REST estableciendo la cabecera HTTP `ibm-mq-rest-gateway-qmgr` al nombre del gestor de colas de pasarela.
4. Incluya en el URL del recurso el nombre del gestor de colas remoto que desee administrar. Por ejemplo, para obtener la lista de colas del gestor de colas remoto `remoteQM`, utilice el URL siguiente:

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

## Resultados

Se devolverá una cabecera `ibm-mq-rest-gateway-qmgr` con la respuesta REST. Esta cabecera especifica qué gestor de colas se utiliza como pasarela.

Si tiene dificultades con el uso de administrative REST API para administrar gestores de colas remotos.

- Compruebe que se está ejecutando el gestor de colas remoto.
- Compruebe que el servidor de mandatos se está ejecutando en el sistema remoto.
- Compruebe que el intervalo de desconexión del canal no ha caducado. Por ejemplo, si se ha iniciado un canal, pero se ha concluido al cabo de un tiempo. Esto es especialmente importante si inicia manualmente los canales.

## Ejemplo

En el ejemplo siguiente, hay tres instalaciones de IBM MQ en dos máquinas. En Machine 1, hay un Installation 1 y un Installation 2. En Machine 2, hay un Installation 3. Se ha configurado un servidor mqweb para Installation 1. Hay un único gestor de colas en cada instalación y dichos gestores de colas están configurados para la administración remota. Es decir, se configuran e inician los siguientes escuchas, canales y colas:

- En el gestor de colas QM1, en Installation 1, en Machine 1:
  - Canal emisor QM1.to.QM2
  - Canal receptor QM2.to.QM1
  - Canal emisor QM1.to.QM3
  - Canal receptor QM3.to.QM1
  - Cola de transmisión QM2
  - Cola de transmisión QM3
  - Un escucha configurado en el puerto 1414
- En el gestor de colas QM2, en Installation 2, en Machine 1:

- Canal emisor QM2.to.QM1
- Canal receptor QM1.to.QM2
- Cola de transmisión QM1
- Un escucha configurado en el puerto 1415
- En el gestor de colas QM3, en Installation 3, en Machine 2:
  - Canal emisor QM3.to.QM1
  - Canal receptor QM1.to.QM3
  - Cola de transmisión QM1
  - El escucha predeterminado

En QM2 se define la cola Qon2 y en QM3 se define la cola Qon3.

Al usuario mquser, definido en ambas máquinas, se le otorga el rol MQWebAdmin en REST API y la autoridad de acceder a las correspondientes colas en cada gestor de colas.

El mandato setmqweb se utiliza para configurar el gestor de colas QM1 como gestor de colas de pasarela predeterminado.

El diagrama siguiente muestra esta configuración:

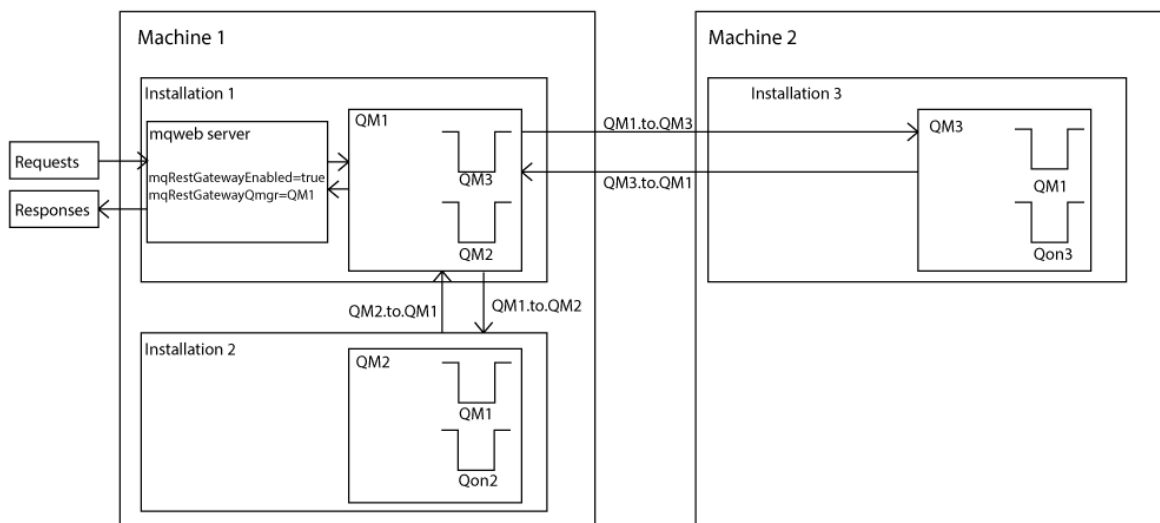


Figura 14. Diagrama de configuración de ejemplo de una administración remota mediante REST API.

Se envía la siguiente solicitud REST al servidor mqweb:

```
GET https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM2/queue?
attributes=general.isTransmissionQueue
```

Se recibe la siguiente respuesta:

```
{
  "queue" :
  [ {
    "general": {
      "isTransmissionQueue": true
    },
    "name": "QM1",
    "type": "local"
  },
  {
    "general": {
      "isTransmissionQueue": false
    }
  }
]
```

```
    "name" : "Qon2",
    "type" : "local"
  }
}
```

Se envía la siguiente solicitud REST al servidor mqweb:

```
GET https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM3/queue?
attributes=general.isTransmissionQueue,general.description
```

Se recibe la siguiente respuesta:

```
{
  "queue" :
  [ {
    "general": {
      "isTransmissionQueue": true,
      "description": "Transmission queue for remote admin."
    },
    "name": "QM1",
    "type": "local"
  },
  {
    "general": {
      "isTransmissionQueue": false,
      "description": "A queue on QM3."
    },
    "name" : "Qon3",
    "type" : "local"
  }
]
```

## V 9.1.0 Indicaciones de fecha y hora de REST API

Cuando administrative REST API devuelve información de fecha y hora, esta se devuelve como Hora universal coordinada (UTC) con un formato fijo.

La fecha y hora se devuelven con el siguiente el formato de indicación de fecha y hora:

```
YYYY-MM-DDTHH:mm:ss:sssZ
```

Por ejemplo, 2012-04-23T18:25:43.000Z, donde la Z indica que el huso horario es la Hora universal coordinada (UTC).

La precisión de esta indicación de fecha y hora no está garantizada. Por ejemplo, si el servidor mqweb no se inicia en el mismo huso horario que el gestor de colas que se especifica en el URL del recurso, puede que la indicación de fecha y hora no sea exacta. Asimismo, si se requieren ajustes de horario de verano, puede que la indicación de fecha y hora no sea precisa.

## Manejo de errores de REST API

REST API informa de los errores devolviendo un código de respuesta HTTP correspondiente, por ejemplo 404 (No encontrado), y una respuesta JSON. Cualquier código de respuesta HTTP que no esté en el rango 200 - 299 se considera un error.

### Formato de respuesta de error

La respuesta está en formato JSON y en codificación UTF-8. Contiene objetos JSON anidados:

- Un objeto JSON externo que contiene una matriz JSON única denominada `error`.
- Cada elemento de la matriz es un objeto JSON que representa información sobre un error. Cada objeto JSON contiene las siguientes propiedades:

#### tipo

Serie.

Tipo de error.

**messageId**

Serie.

Identificador exclusivo del mensaje en formato MQWBnnnnX. Este identificador tiene los siguientes elementos:

**MQWB**

Prefijo que muestra que el mensaje se ha originado en el API REST de MQ.

**nnnn**

Número exclusivo que identifica el mensaje.

**X**

Letra única que indica la gravedad del mensaje:

- I si el mensajes es meramente informativo.
- W si el mensaje es un aviso de un problema.
- E si el mensaje indica que se ha producido un error.
- S si el mensaje indica que se ha producido un error.

**mensaje**

Serie.

Una descripción del error.

**explanation**

Serie.


Una explicación del error.

**action**

Serie.

Una descripción de los pasos que se pueden realizar para resolver el error.

**qmgrName**

 Este campo solo está disponible para z/OS, donde el gestor de colas es miembro del grupo de compartición de colas. Debe haber especificado el parámetro de consulta opcional **commandScope** o el atributo **queueSharingGroupDisposition**.

Serie.

El nombre del gestor de colas que ha experimentado el error.

Este campo no se aplica para messaging REST API.

**completionCode**

Este campo solo está disponible cuando **type** es **pcf**, **java** o **rest**.

Número.

Código de terminación MQ asociado al error.

**reasonCode**

Este campo solo está disponible cuando **type** es **pcf**, **java** o **rest**.

Número.

Código de razón MQ asociado al error.


**exceptions**

Este campo solo está disponible cuando **type** es **java**.

Matriz.

Una matriz de excepciones Java o JMS de cadena. Cada elemento de la matriz de excepciones contiene una matriz de series **stackTrace**.

La matriz de series **stackTrace** contiene los detalles de cada excepción divididos en líneas.

 A partir de IBM MQ 9.1.2, este campo ya no se devuelve.

## Errores con los grupos de compartición de colas

z/OS


En un grupo de compartición de colas, se puede especificar un parámetro de consulta opcional de **commandScope** en determinados mandatos. Este parámetro permite propagar el mandato a otros gestores de colas del grupo de compartición de colas. Cualquiera de estos mandatos puede fallar independientemente, por lo que algunos mandatos terminarán satisfactoriamente y otros fallarán para el grupo de compartición de colas.

En los casos en los que un mandato falla parcialmente, se devuelve un código de error HTTP 500. Para cada gestor de colas que genera un error, se devuelve información sobre ese error como un elemento en la matriz JSON `error`. Para cada gestor de colas que ejecuta correctamente el mandato, se devuelve el nombre del gestor de colas como un elemento en una matriz JSON `success`.

### Ejemplos

- El ejemplo siguiente muestra la respuesta de error a un intento de obtener información sobre un gestor de colas que no existe:

```
"error": [
  {
    "type": "rest",
    "messageId": "MQWB0009E",
    "message": "MQWB0009E: Could not query the queue manager 'QM1'",
    "explanation": "The MQ REST API was invoked specifying a queue manager name which cannot be located.",
    "action": "Resubmit the request with a valid queue manager name or no queue manager name, to retrieve a list of queue managers. "
  }
]
```

-  El ejemplo siguiente muestra la respuesta de error a un intento de suprimir una cola en un grupo de compartición de colas que no existe para algunos gestores de colas:

```
"error" : [
  {
    "type": "rest",
    "messageId": "MQWB0037E",
    "message": "MQWB0037E: Could not find the queue 'missingQueue' - the queue manager reason code is 3312 : 'MQRCCF_UNKNOWN_OBJECT_NAME'",
    "explanation": "The MQ REST API was invoked specifying a queue name which cannot be located.",
    "action": "Resubmit the request with the name of an existing queue, or with no queue name to retrieve a list of queues.",
    "qmgrName": "QM1"
  },
  {
    "type": "rest",
    "messageId": "MQWB0037E",
    "message": "MQWB0037E: Could not find the queue 'missingQueue' - the queue manager reason code is 3312 : 'MQRCCF_UNKNOWN_OBJECT_NAME'",
    "explanation": "The MQ REST API was invoked specifying a queue name which cannot be located.",
    "action": "Resubmit the request with the name of an existing queue, or with no queue name to retrieve a list of queues.",
    "qmgrName": "QM2"
  }
],
"success" : [{"qmgrName": "QM3"}, {"qmgrName": "QM4"}]
```

### Errores con solicitudes MFT

Si los servicios MFT REST API no están habilitados, e invoca MFT REST API, recibirá la excepción siguiente:

```
{"error": [{"
  "action": "Enable the Managed File Transfer REST API and resubmit the request.",
  "completionCode": 0,
```

```
"explanation": "Managed File Transfer REST calls are not permitted as the service is disabled.",
"message": "MQWB0400E: Managed File Transfer REST API is not enabled.",
"msgId": "MQWB0400E",
"reasonCode": 0,
"type": "rest"
}}}
```

Si los servicios MFT REST API están habilitados y el gestor de colas de coordinación no está establecido en el archivo `mqwebuser.xml`, recibirá la siguiente excepción:

```
{"error": [{
  "action": "Set the coordination queue manager name and restart the mqweb server.",
  "completionCode": 0,
  "explanation": "Coordination queue manager name must be set before using Managed File Transfer REST services.",
  "message": "MQWB0402E: Coordination queue manager name is not set.",
  "msgId": "MQWB0402E",
  "reasonCode": 0,
  "type": "rest"
}]}
```

## V 9.1.0 Descubrimiento de REST API

La Documentación para REST API está disponible en IBM Documentation y en formato Swagger. Swagger es un enfoque utilizado comúnmente para documentar las API REST. La documentación de Swagger para REST API puede visualizarse habilitando la característica de descubrimiento de API en el servidor mqweb.

### Antes de empezar

**Importante:** La característica de descubrimiento de API se ha estabilizado y todavía puede utilizar esta característica. En la actualidad, IBM MQ no da soporte al uso de la característica `mpOpenAPI`.

Debe habilitar la seguridad para el servidor mqweb para ver la documentación Swagger utilizando el descubrimiento de API. Si desea más información sobre los pasos que son necesarios para habilitar la seguridad, consulte [Seguridad de IBM MQ Console y REST API](#).

### Procedimiento

1. Localice el archivo `mqwebuser.xml` en uno de los directorios siguientes:

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

Donde `WLP_user_directory` es el directorio que se especificó cuando se ejecutó el script

`V 9.1.0` `crtmqweb` para crear la definición de servidor mqweb.

2. Añada el XML adecuado al archivo `mqwebuser.xml`:

- Si las etiquetas `<featureManager>` existen en el archivo `mqwebuser.xml`, añada el XML siguiente dentro de las etiquetas `<featureManager>`:  
`<feature>apiDiscovery-1.0</feature>`
- Si las etiquetas `<featureManager>` no existen en el archivo `mqwebuser.xml`, añada el XML siguiente dentro de las etiquetas `<server>`:

```
<featureManager>
  <feature>apiDiscovery-1.0</feature>
</featureManager>
```

3. Vea la documentación Swagger utilizando uno de los métodos siguientes:

- Visualice una página web que pueda examinar y pruebe REST API especificando el URL siguiente en un navegador:

`https://host:port/ibm/api/explorer`

Además de autenticar cada solicitud, debe incluir una cabecera `ibm-mq-rest-csrf-token` para cada solicitud POST, PATCH o DELETE. El contenido de esta cabecera puede ser cualquier serie, incluido el espacio en blanco.

Esta cabecera de solicitud se utiliza para confirmar que las credenciales para autenticar la solicitud las utiliza el propietario de las credenciales. Es decir, se utiliza la señal para impedir ataques de falsificación de solicitudes entre sitios.

- Recupere un único documento Swagger 2 que describe completamente REST API emitiendo HTTP GET al URL siguiente:

`https://host:port/ibm/api/docs`

Este documento puede utilizarse para aplicaciones donde desea navegar mediante programa en las API disponibles.

#### **host**

Especifica el nombre de host o la dirección IP en la que está disponible REST API.

El valor predeterminado es `localhost`.

#### **port**

Especifica el número de puerto HTTPS que utiliza administrative REST API.

El valor predeterminado es `9443`.

Si el nombre de host o número de puerto cambia respecto al valor predeterminado, puede determinar los valores correctos a partir del URL de REST API. Utilice el mandato `dspmweb status` para ver el URL.

**V 9.1.0**

## **Soporte multilingüístico de REST API**

REST API soporta, con determinadas cualificaciones, la capacidad de especificar idiomas nacionales como parte de una solicitud HTTP.

### **En segundo plano**

Las cabeceras HTTP permiten que se especifique un comportamiento concreto en solicitudes y que se proporcione información adicional en respuestas.

Se incluye en las cabeceras HTTP la capacidad de solicitar que dicha información se devuelva en un idioma nacional. REST API respeta esta cabecera siempre que sea posible.

### **Especificar un idioma nacional**

En la cabecera `ACCEPT-LANGUAGE` HTTP, se puede proporcionar uno o más códigos de idioma. De forma opcional, puede asociar una clasificación con los códigos, lo que le permite la especificación de una lista ordenada por preferencia. Esta página tiene un debate útil del principio.

REST API respeta esta cabecera, seleccionando un idioma de la cabecera `ACCEPT-LANGUAGE` y devolviendo mensajes en dicho idioma. Cuando la cabecera `ACCEPT-LANGUAGE` no contiene ningún idioma que REST API pueda soportar, los mensajes se devuelven en un idioma predeterminado. Este idioma predeterminado corresponde al entorno local predeterminado del servidor web de REST API.

La sección “¿Qué datos se traducen?” en la [página 89](#) explica qué datos se traducen.

### **Indicar el idioma aplicable en las respuestas**

La cabecera `HTTP CONTENT-LANGUAGE` en las respuestas de REST API indica el idioma en el cual se devuelven los mensajes.



## ¿Qué datos se traducen?

Los mensajes de error e informativos se traducen, otro texto no.

- Los datos que se devuelven de un gestor de colas no se traducen, por ejemplo, en el caso de ejecutar un mandato MQSC mediante REST API, las respuestas del gestor de colas están en el entorno local del gestor de colas.
- La documentación generada (Swagger) para REST API, tal como se expone mediante la característica `apiDiscovery`, está en inglés.

## ¿Qué idiomas están soportados?

Además del inglés, los mensajes de error e informativos de REST API están traducidos a los idiomas siguientes.

### Chino (Simplificado)

Indicado por el código de idioma `zh_CN`

### Chino (Tradicional)

Indicado por el código de idioma `zh_TW`

### Checo

Indicado por el código de idioma `cs`

### Francés

Indicado por el código de idioma `fr`

### Húngaro

Indicado por el código de idioma `hu`

### Italiano

Indicado por el código de idioma `it`

### Japonés

Indicado por el código de idioma `ja`

### Coreano

Indicado por el código de idioma `ko`

### Polaco

Indicado por el código de idioma `pl`

### Portugués (Brasil)

Indicado por el código de idioma `pt_BR`

### Ruso

Indicado por el código de idioma `ru`

### Español

Indicado por el código de idioma `es`

## Ejemplos

En los ejemplos, el servidor web tiene un entorno local inglés.

### Especificación de un único idioma soportado

En las cabeceras de petición, `ACCEPT-LANGUAGE` está establecido en `fr`. Este valor especifica que el francés es el idioma preferido para el texto traducible.

En las cabeceras de respuesta, `CONTENT-LANGUAGE` está establecido en `fr`. Este valor indica que los mensajes de error e información en la respuesta están en francés.

### Especificación de una lista de idiomas

En las cabeceras de petición, `ACCEPT-LANGUAGE` está establecido en `am, fr`. Este valor especifica que el amhárico y el francés son idiomas aceptables para texto traducible y que el amhárico es el idioma preferido para el texto traducible.

En las cabeceras de respuesta, CONTENT-LANGUAGE está establecido en fr. Este valor indica que los mensajes de error e informativos en la respuesta están en francés, ya que REST API no soporta el amhárico.

#### **Especificación de un único idioma no soportado**

En las cabeceras de petición, ACCEPT-LANGUAGE está establecido en am. Este valor especifica que el amhárico es el idioma preferido para el texto traducible.

En las cabeceras de respuesta, CONTENT-LANGUAGE está establecido en en. Este valor indica que los mensajes de error e informativos en la respuesta están en inglés, ya que REST API no soporta el amhárico.

## **V 9.1.0 Versiones de REST API**

En IBM MQ 9.1, la REST API está en la versión 1. A partir de IBM MQ 9.1.5, el REST API está en la versión 2. Este número de versión forma parte del URL base para solicitudes REST. Por ejemplo, `https://localhost:9443/ibmmq/rest/v2/admin/installation`. El número de versión se utiliza para aislar los clientes de los cambios en la REST API que se podrían introducir en futuros releases.

Algunos cambios que se han introducido en la REST API podrían cambiar la función de la REST API existente como, por ejemplo, que los clientes que utilizan la REST API podrían tener que actualizarse. Para impedir que estos cambios fueren la actualización de los clientes, el número de versión de REST API se aumenta y la función existente se estabiliza en el número anterior. La nueva función que podría cambiar la función existente se añade a la REST API en el nuevo número de versión. Por lo tanto, los clientes pueden seguir utilizando la REST API en la versión anterior sin actualizarse.

Los cambios de REST API que podrían dar como resultado que sea necesario una actualización de cliente incluyen los cambios siguientes:

- La eliminación del soporte para un atributo existente en el JSON que se envía a, o devuelve de, la REST API.
- La eliminación de un URL, verbo HTTP o cabecera. Por ejemplo, si se renombra un URL o una cabecera, o si se utiliza un verbo diferente.
- La adición de un nuevo atributo JSON obligatorio a los datos que se envían a un URL existente.
- La adición de una nueva cabecera HTTP obligatoria a los datos que se envían a un URL existente.
- La adición de un nuevo parámetro de consulta obligatorio a un URL existente.

Cuando este tipo de cambio se introduce en la función de REST API que existía en un release de Long Term Support (LTS), el número de versión de la REST API se aumenta para el primero de estos cambios. Cualquier cambio posterior que se haya realizado en un release de Continuous Delivery (CD) que pueda requerir cambios en los clientes que utilizan la REST API utiliza el nuevo número de versión.

Este número de versión sigue siendo el mismo a lo largo de todos los releases posteriores de CD hasta el siguiente release de LTS. Por lo tanto, el número de versión aumenta como máximo una vez entre los releases de LTS.

Cuando el número de versión se aumenta, la función de REST API existente se estabiliza en el número de versión antiguo. Es decir, la función REST API existente que estaba disponible en el release de LTS sigue estando disponible en el número de versión antigua, pero no se realizan más cambios en esa versión. Cualquier función nueva que se añada a la REST API se añade a la nueva versión de la REST API. Sin embargo, no se garantiza que las adiciones que se realizan en la REST API en los releases de CD antes del aumento de versión estén incluidas en la versión más antigua de la REST API.

Los clientes existentes pueden seguir utilizando la REST API en el número de versión antiguo sin necesitar ningún cambio. Las versiones más antiguas de la REST API podrían estar en desuso y, finalmente, haber sido eliminadas.

Algunos cambios no requieren cambios en los clientes que utilizan la REST API. Estos cambios no generan un aumento del número de versión. Por lo tanto, asegúrese de que cualquier cliente que utiliza la REST API no es necesario que se actualice cuando se introduzcan estos tipos de cambios. Estos cambios en la REST API podrían incluir los cambios siguientes:

- Adición de un nuevo atributo JSON a los datos existentes que se devuelven de la REST API.
- Adición de un nuevo URL.
- Adición de un nuevo verbo HTTP a un URL existente.
- Adición de un nuevo código de estado a un URL existente.
- Adición de nuevos atributos JSON opcionales a los datos que se envían a un URL existente.
- Adición de nuevos parámetros de consulta en un URL existente.
- Adición de nuevas cabeceras a los datos que se envían a un URL existente.
- Devolución de nuevas cabeceras de la REST API.

## Cambios en la nueva función de la API REST Continuous Delivery

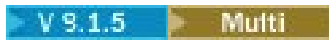
Para la nueva función de REST API que se ha añadido en un release de CD, los cambios realizados en esta nueva función que podrían necesitar cambios en los clientes de REST API no aumentan el número de versión. Es decir, la nueva función puede cambiar antes del siguiente release de LTS sin aumentar el número de versión. Cuando la función está incluida en un release de LTS, los cambios posteriores que podrían requerir cambios en clientes de REST API no aumentan el número de versión.

### Ejemplo

1. En el release X de LTS, la REST API está en la versión 1.
2. En el release CD X.0.1, se ha añadido el soporte para un nuevo URL. Este cambio no requiere cambios en clientes que utilizan la REST API. Por lo tanto, la REST API permanece en la versión 1.
3. En CD X.0.2, se ha añadido soporte para un nuevo URL. Este cambio no requiere cambios en los clientes que utilizan la API REST. Por lo tanto, la REST API permanece en la versión 1.
4. En LTS release Y, la REST API está en la versión 1.
5. En CD release Y.0.1, se renombra un URL existente. Este cambio podría requerir cambios en los clientes que utilizan la REST API. Por consiguiente, se crea una nueva versión de REST API como versión 2. El URL renombrado se incluye en la versión 2 de REST API, junto con toda la función existente. Cualquier nueva función que se añada a REST API se añade a la versión 2. La versión 1 permanece estabilizada en el nivel del release Y de LTS.
6. En CD release Y.0.2, se renombra otro URL existente. Como la versión ya se ha incrementado en CD release Y, REST API permanece en la versión 2. La versión 1 permanece estabilizada en el nivel del release Y de LTS.
7. En LTS release Z, REST API permanece en la versión 2. La versión 1 permanece estabilizada en el nivel del release Y de LTS.

## Administración utilizando una consola web

Puede realizar tareas de administración básicas utilizando una consola web.

 Desde IBM MQ 9.1.5, una nueva consola web, conocida como New Web Console, está disponible en IBM MQ for Multiplatforms, consulte [Visita rápida de New Web Console](#). Para IBM MQ Appliance y IBM MQ for z/OS, debe seguir utilizando la consola web original, que ahora se conoce como Dashboard Web Console, consulte [“Dashboard Web Console”](#) en la [página 97](#).

Si lo desea, puede continuar utilizando el Dashboard Web Console en multiplataformas (consulte [“Conmutación de tipos de consola en IBM MQ for Multiplatforms”](#) en la [página 97](#)).

**Nota:** No inhabilite el servidor de mandatos en ninguno de los gestores de colas cuando utilice una consola web. Si el servidor de mandatos está inhabilitado para un gestor de colas, la consola web no responde y tiene grandes retardos en el proceso de mandatos. Los mandatos que se emitan al gestor de colas mientras el servidor de mandatos esté inhabilitado excederán el tiempo de espera.

### Tareas relacionadas

[Rastreo de la IBM MQ Console y la REST API](#)

## V 9.1.5 **Cómo empezar con la consola web**

Cómo empezar rápidamente con la consola web. Estas instrucciones se aplican tanto a la New Web Console como a la Dashboard Web Console.

### **Antes de empezar**

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar el mandato **dspmqweb**:

- ▶ **z/OS** En z/OS, debe tener autorización para ejecutar el mandato **dspmqweb** y acceso de escritura en el archivo `mqwebuser.xml`.
- ▶ **Multi** En todos los demás sistemas operativos, debe ser un usuario con privilegios.
- ▶ **IBM i** En IBM i, los mandatos deben estar en ejecución en QSHELL.

### **Procedimiento**

1. Si el servidor mqweb todavía no está configurado para utilizarlo en la consola web, configure el servidor mqweb.

Para obtener más información sobre la creación de una configuración básica para el servidor mqweb con un registro básico, consulte [Configuración básica para el servidor mqweb](#).

2. ▶ **z/OS**  
En z/OS, establezca la variable de entorno `WLP_USER_DIR` para que pueda utilizar el mandato **dspmqweb**. Establezca la variable para que apunte a la configuración del servidor mqweb especificando el mandato siguiente.

```
export WLP_USER_DIR=WLP_user_directory
```

donde `WLP_user_directory` es el nombre del directorio que se pasa a `crtmqweb`. Por ejemplo:

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

Si desea más información, consulte [Creación del servidor mqweb](#).

3. Determine el URI de la consola web especificando el mandato siguiente:

```
dspmqweb status
```

El mandato genera una salida similar a la siguiente:

```
MQWB1124I: Server 'mqweb' is running.  
URLS:  
https://localhost:9443/ibmmq/rest/v1/  
https://localhost:9443/ibmmq/console/
```

El URI de la consola web finaliza con el sufijo `console/`.

4. Conéctese a la consola web especificando el URL desde “3” en la [página 92](#) en un navegador.

El navegador podría generar una excepción de seguridad porque el certificado predeterminado que se proporciona con el servidor no es un certificado de confianza. Elija continuar con la consola web.

5. Inicie sesión en la consola. Utilice el nombre de usuario `mqadmin` y la contraseña `mqadmin`.

### **Qué hacer a continuación**

- De forma predeterminada, la consola web utiliza la autenticación basada en señal para autenticar a los usuarios. También puede utilizar la autenticación de certificado de cliente. Para obtener más información, consulte [Utilización de la autenticación de certificados de cliente con REST API y la consola web](#).

- Obtenga más información sobre el uso de la consola web: [“Cómo empezar con la consola web”](#) en la página 92 o [“Dashboard Web Console”](#) en la página 97.

## **Restricciones en z/OS**

Se aplican las siguientes restricciones cuando se utiliza IBM MQ Console para gestionar gestores de colas en z/OS.

- No pueden crearse, suprimirse, iniciarse ni detenerse gestores de colas en z/OS.
- Los iniciadores de canal en z/OS no pueden iniciarse ni detenerse, y no se muestra el estado del iniciador de canal.
- Los escuchas no pueden visualizarse ni administrarse.
- Los mandatos para iniciar, ejecutar ping, resolver y restablecer el canal solo pueden emitirse con CHLDISP(DEFAULT).
- Los nuevos objetos solo pueden crearse con QSGDISP(QMGR).
- Los objetos definidos con QSGDISP(GROUP) no pueden visualizarse ni gestionarse.
- La seguridad del gestor de colas no puede gestionarse.
- El uso de recursos de sistema no se puede supervisar.

### **Conceptos relacionados**

[“Administración utilizando una consola web”](#) en la página 91

Puede realizar tareas de administración básicas utilizando una consola web.

[Trabajar con gestores de colas locales](#)

## **Restricciones en IBM MQ for Multiplatforms**

Se aplican las siguientes restricciones cuando se utiliza IBM MQ Console para gestionar gestores de colas en IBM MQ for Multiplatforms.

- No puede utilizar IBM MQ Console para trabajar con canales AMQP.
- No puede utilizar IBM MQ Console para trabajar con canales MQTT.

### **Conceptos relacionados**

[“Administración utilizando una consola web”](#) en la página 91

Puede realizar tareas de administración básicas utilizando una consola web.

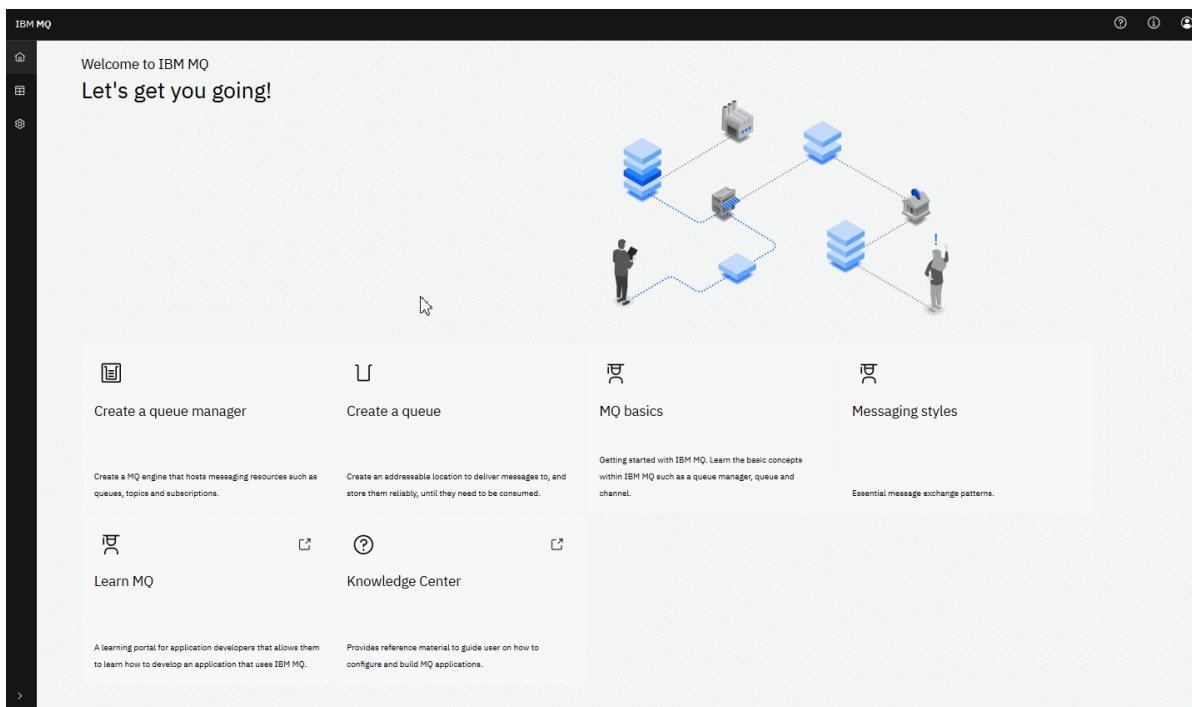
[Trabajar con gestores de colas locales](#)

## **Visita rápida de la New Web Console**

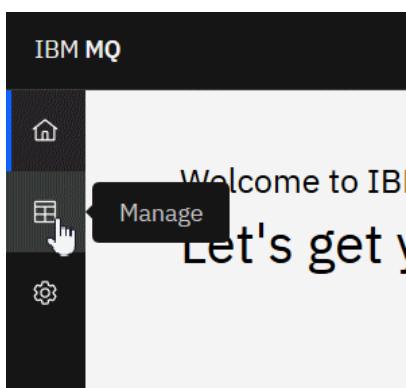
IBM MQ 9.1.5 tiene una nueva consola, denominada la New Web Console, con un aspecto nuevo.

Esta visita rápida tiene el objetivo de familiarizarse con la New Web Console. Si desea continuar utilizando la consola existente ( Dashboard Web Console), puede volver a ello, consulte [Conmutación de tipos de consola en IBM MQ for Multiplatforms](#).

La primera vez que inicia sesión en la consola se le dirige a la página de destino. Desde aquí puede elegir crear un gestor de colas o una cola, navegar a algunos temas de formación o abrir la información del producto IBM MQ en IBM Documentation.



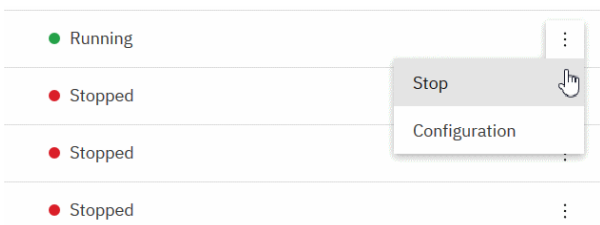
O bien, puede pulsar el icono gestionar para empezar a gestionar los objetos de IBM MQ inmediatamente.



La vista de gestión muestra inicialmente los gestores de colas y su estado actual. También puede crear nuevos gestores de colas.

Queue manager name ↑	Version	Status
MpgQM2	9.1.5.0	Running
MyPortQmgr	9.1.5.0	Stopped
QM1	9.1.5.0	Stopped
qmanotherone	9.1.5.0	Stopped
qmq	9.1.5.0	Stopped
qnd	9.1.5.0	Stopped
qne	9.1.5.0	Stopped
RPilotFinalTest	9.1.5.0	Running
sdwe	9.1.5.0	Stopped
Steves_qm	9.1.5.0	Stopped

Cada gestor de colas tiene un menú que le permite detener o configurar un gestor de colas en ejecución, o iniciar o suprimir un gestor de colas detenido.



Los registros de autorizaciones, los objetos de información de autenticación y los registros de autenticación de canal para el gestor de colas se pueden encontrar en la pestaña **Seguridad** de la página **Configuración** del gestor de colas, donde puede crear y añadir unos nuevos.


Pulse el nombre de un gestor de colas en ejecución para abrir su panel de instrumentos.

The screenshot shows the IBM MQ Queue Manager interface for 'MpgQM'. It features a navigation menu on the left with options for Queues, Topics, Subscriptions, and Communication. The main area displays three capacity status cards: '1 Full capacity' (1 user-defined, 0 system), '2 Greater than 50% capacity' (1 user-defined, 1 system), and '61 Less than 1% capacity' (15 user-defined, 46 system). Below these is a table of queues with columns for Name, Type, Depth %, and Maximum depth. A 'Create +' button is visible in the top right of the table area.


Name ↑	Type	Depth %	Maximum depth
AMQ.MQEXPLORER.5E68C78B20805304	Local	0%	0/5000
Queueue1	Local	0%	0/1
Queueue2	Local	0%	0/5000
Queueue3	Local	0%	0/5000
Queueue4	Local	0%	0/5000

Desde el panel de instrumentos del gestor de colas, puede completar las acciones siguientes:


En la pestaña **Colas**:

- Crear colas nuevas
- Configurar las colas existentes 
- Pulsar un nombre de cola para ver los mensajes existentes y crear unos nuevos



En la pestaña **Temas**:

- Crear temas nuevos
- Configurar los temas existentes 
- Pulse un nombre de tema para ver las suscripciones coincidentes

En la pestaña **Suscripciones**:

- Crear suscripciones nuevas gestionadas o sin gestionar
- Configurar las suscripciones existentes 

En la pestaña **Comunicaciones**:

- Escuchas:
  - Iniciar, detener y configurar las escuchas 
  - Crear escuchas nuevas
- Canales de gestor de colas:
  - Iniciar, detener, hacer ping y configurar canales 
  - Crear nuevos canales
  - Restablecer canales (desde el elemento de menú **Avanzado**)
  - Resolver mensajes pendientes en canales (desde el elemento de menú **Avanzado**)



- Canales de la aplicación:

- Iniciar, detener, hacer ping y configurar canales
- Crear nuevos canales
- Restablecer canales (desde el elemento de menú **Avanzado**)
- Resolver mensajes pendientes en canales (desde el elemento de menú **Avanzado**)

## V 9.1.5 Multi Conmutación de tipos de consola en IBM MQ for Multiplatforms

Puede optar por seguir utilizando el Dashboard Web Console heredado en lugar de utilizar la New Web Console en IBM MQ for Multiplatforms.

### Acerca de esta tarea

De forma predeterminada, está habilitada la New Web Console en multiplataformas. Puede cambiar a utilizar la consola web original (ahora se conoce como Dashboard Web Console) editando un valor en el archivo `mqwebuser.xml`. En UNIX y Linux, `mqwebuser.xml` se encuentra en el directorio `var/mqm/web/installations/installationName/servers/mqweb`. En Windows, `mqwebuser.xml` se encuentra en el directorio `MQ_DATA_PATH\web\installations\installationName\servers\mqweb`.

### Procedimiento

- Para cambiar a utilizar Dashboard Web Console:
  - a) En el archivo `mqwebuser.xml`, localice la propiedad denominada `mqConsoleEarName`.
  - b) Cambie el valor de propiedad para establecerlo en `com.ibm.mq.console`:

```
<variable name="mqConsoleEarName" value="com.ibm.mq.console"/>
```

- Para volver a utilizar la New Web Console:
  - a) En el archivo `mqwebuser.xml`, localice la propiedad denominada `mqConsoleEarName`.
  - b) Cambie el valor de propiedad para establecerlo en `com.ibm.mq.webconsole`:

```
<variable name="mqConsoleEarName" value="com.ibm.mq.webconsole"/>
```

### Resultados

Cuando especifica el URL de consola web en un navegador, aparece el tipo seleccionado de la consola web.

**Nota:** los únicos valores válidos para la propiedad `mqConsoleEarName` son `com.ibm.mq.webconsole` y `com.ibm.mq.console`. Cualquier otro valor causa el error siguiente:

```
00000107 com.ibm.ws.app.manager.AppMessageHelper W CWWKZ0014W: The application com.ibm.mq.console could not be started as it could not be found at location C:/Program Files/IBM/MQ/web/mq/apps/com.ibm.mq.invalid.ear.
```

## V 9.1.5 Dashboard Web Console

A partir de IBM MQ 9.1.5, la consola web disponible en las versiones anteriores se conoce como Dashboard Web Console.

Está disponible una nueva consola web, llamada New Web Console, para IBM MQ for Multiplatforms (consulte [Visita rápida de la New Web Console](#)). Los usuarios de IBM MQ for z/OS deben seguir utilizando el Dashboard Web Console, que se describe en los temas siguientes.

Los usuarios de IBM MQ for Multiplatforms pueden optar por seguir utilizando el Dashboard Web Console, consulte [“Conmutación de tipos de consola en IBM MQ for Multiplatforms”](#) en la página 97.

## **V 9.1.0** Trabajar con gestores de colas locales

Puede utilizar el widget de gestor de colas local en IBM MQ Console para crear, configurar y controlar los gestores de colas locales.

### Acerca de esta tarea

**Multi** El widget de gestor de colas local enumera los gestores de colas locales añadidos a la instalación de IBM MQ desde la que se está ejecutando IBM MQ Console. Los gestores de colas asociados a distintas instalaciones de IBM MQ en el mismo sistema no se enumeran.

**z/OS** En z/OS, el widget de gestor de colas local enumera los gestores de colas que están en la misma versión que IBM MQ Console y están definidos en el sistema en el que se está ejecutando MQ Console. Los gestores de colas de una versión diferente a la de MQ Console no se enumeran.

Puede seleccionar gestores de colas individuales en la lista para trabajar con ellos.

**Nota:** **V 9.1.0** IBM MQ Console no admite los gestores de colas de datos replicados.

Puede añadir un widget de gestor de colas local al panel de instrumentos pulsando **añadir widget**

**Add widget**

. A continuación, seleccione **Gestores de colas locales**.

Puede cambiar el nombre del widget después de haberlo creado. Pase el puntero del ratón por encima


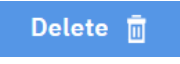
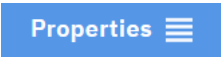


de la barra de título y aparecerá el icono de cambio de nombre **Rename widget**. Pulse el icono, especifique un nombre nuevo en la ventana Renombrar widget y pulse **Renombrar**.

**z/OS** No puede crear, iniciar, detener ni suprimir gestores de colas en z/OS.

### Procedimiento

- Para crear un nuevo gestor de colas local:
  - a) Pulse el icono de creación **Create +** en la barra de herramientas del widget del gestor de colas local.
  - b) Escriba un nombre para el nuevo gestor de colas. El nombre puede tener hasta 48 caracteres. Los caracteres válidos son letras y números y los caracteres ":", "/", "\_ y %".
  - c) Opcional: Especifique un puerto TCP/IP disponible para la escucha del gestor de colas. El número de puerto no debe exceder 65535.
  - d) Pulse **Crear**. Se crea y se inicia el nuevo gestor de colas.
- Para iniciar un gestor de colas local:
  - a) Seleccione el gestor de colas que desea iniciar de la lista en el widget de gestor de colas local.
  - b) Pulse el icono de inicio **Start ▶** en la barra de herramientas del widget del gestor de colas local.
- Para detener un gestor de colas local:
  - a) Seleccione el gestor de colas que desea detener de la lista en el widget de gestor de colas local.

- b) Pulse el icono Detener  en la barra de herramientas del widget del gestor de colas local.
- c) Confirme que desea detener el gestor de colas pulsando **Detener**.
- Para suprimir un gestor de colas local:
  - a) Seleccione el gestor de colas que desea suprimir de la lista en el widget de gestor de colas local.
  - b) Si el gestor de colas está en ejecución, deténgalo.
  - c) Pulse el icono Suprimir  en la barra de herramientas del widget del gestor de colas local.
  - d) Confirme que desea suprimir el gestor de colas pulsando **Suprimir**. Se suprimen el gestor de colas y todos los objetos asociados.
- Para ver y editar las propiedades de un gestor de colas local:
  - a) Asegúrese de que el gestor de colas está en ejecución, y selecciónelo en la lista del gestor de colas.
  - b) Pulse el icono de propiedades  en la barra de herramientas del widget del gestor de colas local. O bien, efectúe una doble pulsación en el gestor de colas.
  - c) Consulte las propiedades y edítelas según sea necesario. Si el recuadro de texto de propiedad está inhabilitado, la propiedad es de sólo lectura o solo se puede editar desde la línea de mandatos. Para obtener información sobre una propiedad, puede ver la información de la propiedad en [Propiedades del gestor de colas](#) en la IBM MQ Explorer documentación.
- Para renovar la seguridad para el gestor de colas local:
  - a) Asegúrese de que el gestor de colas local está en ejecución y selecciónelo en la lista del gestor de colas.
  - b) Seleccione ... > **Renovar seguridad**
  - c) Seleccione la seguridad del gestor de colas que desea renovar:
    - Seleccione **Servicio de autorización** para renovar la lista de autorizaciones que está retenida internamente por el componente de servicio de autorización.
    - Seleccione **Autenticación de conexión** para renovar la vista almacenada en memoria caché de la configuración para la autenticación de conexión.
    - Seleccione **SSL** para renovar la vista almacenada en memoria caché del repositorio de claves SSL o TLS. Esta opción también renueva las ubicaciones de los servidores LDAP que se utilizan para las listas de revocación de certificados y los parámetros de hardware de cifrado.
- Para trabajar con los registros de autorizaciones del gestor de colas local:
  - a) Asegúrese de que el gestor de colas local está en ejecución y selecciónelo en la lista del gestor de colas.
  - b) Seleccione una de las opciones siguientes:
    - Seleccione ... > **Gestionar registros de autorización** para trabajar con los registros de autorización para el gestor de colas y especificar qué acciones pueden realizar los grupos de usuarios.
    - Seleccione ... > **Gestionar registros de autorización de creación** para trabajar con los registros de autorización de creación para el gestor de colas y especificar qué objetos pueden crear los grupos de usuarios en ese gestor de colas.
- Para crear automáticamente una pestaña de panel de instrumentos para un gestor de colas local:
  - a) Seleccione el gestor de colas en el widget de gestor de colas local.
  - b) Seleccione ... > **Añadir nueva pestaña del panel de instrumentos**  
Se crea una nueva pestaña del panel de instrumentos. La pestaña tiene el nombre del gestor de colas.

- Para filtrar la lista de gestores de colas locales:
  - a) Escriba el texto de filtro en el recuadro de búsqueda.
  - b) Para detener el filtrado, suprima el texto del recuadro de búsqueda.

## Trabajar con objetos de IBM MQ

Puede utilizar los widgets de objeto de IBM MQ en IBM MQ Console para trabajar con los diferentes tipos de objeto de IBM MQ.



### Acerca de esta tarea

Cada widget de objeto de IBM MQ contiene objetos asociados a un gestor de colas específico. Puede añadir los siguientes tipos de widgets de objeto de IBM MQ al panel de instrumentos:

- Widget de colas
- Widget de temas
- Widget de escuchas
- Widget de canales
- Widget de canales de conexión cliente
- Widget de información de autenticación
- Widget de suscripciones

### Procedimiento

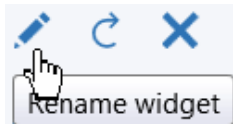
- Para crear un widget de objeto de IBM MQ:

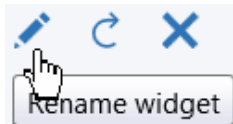
- a) Pulse **Añadir widget**  .
- b) Seleccione el gestor de colas adecuado de la lista.
- c) Pulse el nombre del tipo de widget de objeto que desea crear.

- 

Para cambiar el nombre del widget de objeto:

- a) Pase el puntero del ratón por encima de la barra de título y aparecerá el icono de cambio de






nombre . Pulse el icono.

- b) Especifique un nombre nuevo en la ventana Renombrar widget y pulse **Renombrar**.

- 

Para configurar el widget de objeto de IBM MQ:

- a) Pulse el icono configurar  en la barra de título del widget.
  - b) Opcional: Especifique el gestor de colas para el que se van a visualizar los objetos de IBM MQ.
  - c) Opcional: Especifique si los objetos del sistema se van a mostrar u ocultar.
  - d) Pulse **Guardar**.
- Para filtrar los objetos que se visualizan en el widget:
    - a) Escriba el texto de filtro en el recuadro de búsqueda.
    - b) Para detener el filtrado, suprima el texto del recuadro de búsqueda.
  - Para renovar el contenido del widget, pulse el icono renovar   en la barra de título del widget.

- Para eliminar el widget, pulse el icono eliminar   en la barra de título del widget.

## **Trabajar con colas**

Puede utilizar el widget de colas en IBM MQ Console para mostrar las colas que existen para un gestor de colas específico. A continuación, puede añadir y suprimir colas, añadir y borrar mensajes en una cola, examinar mensajes, ver y establecer las propiedades de una cola y gestionar los registros de autorizaciones de una cola.

### Antes de empezar








Debe crear un widget de colas antes de poder utilizarlo. Para obtener más información sobre cómo crear widgets de objeto de IBM MQ, consulte [“Trabajar con objetos de IBM MQ” en la página 100.](#)




### Acerca de esta tarea


El widget de colas enumera las colas que existen para un gestor de colas específico. Puede seleccionar colas individuales de la lista con las que trabajar.

 No puede ver ni editar registros de autorizaciones de colas en z/OS.

### Procedimiento

- Para añadir una cola:
  - a) Pulse el icono de creación   en la barra de herramientas del widget de colas.
  - b) Escriba un nombre para la cola. Los caracteres válidos son letras y números y los caracteres ":", "/", "\_", y "%".
  - c) Seleccione el tipo de cola que desea añadir.
  - d) Pulse **Crear**. Se crea la nueva cola.
- Para poner mensajes en una cola:
  - a) Seleccione la cola a la que desea añadir mensajes de la lista en el widget de colas. No puede seleccionar una cola de modelo.
  - b) Pulse el icono de colocación de mensaje   en la barra de herramientas del widget de colas.
  - c) Especifique el mensaje que desea transferir a la cola.
  - d) Pulse **Transferir**.
- Para borrar mensajes de una cola:
  - a) Seleccione la cola local de la que desea borrar mensajes de la lista en el widget de colas.
  - b) Seleccione  **... > Borrar cola**.
  - c) Confirme que desea borrar la cola pulsando **Borrar cola**.
- Para examinar mensajes en una cola:
  - a) Seleccione la cola local o alias que desea examinar de la lista en el widget de colas.
  - b) Pulse el icono Examinar   en la barra de herramientas del widget de colas. Se abre la ventana para examinar mensaje, mostrando los mensajes en la cola.
- Para suprimir una cola:
  - a) Seleccione la cola que desea suprimir de la lista en el widget de colas.

- b) Pulse el icono Suprimir  en la barra de herramientas del widget de colas.
  - c) Opcional: Si la cola tiene mensajes, confirme que se puede borrar la cola pulsando **Borrar cola**.
  - d) Confirme que desea suprimir la cola pulsando **Suprimir**. Se suprime la cola.
- Para ver y editar las propiedades de una cola:
    - a) Seleccione la cola en el widget de colas.
    - b) Pulse el icono de propiedades  en la barra de herramientas del widget de colas. O bien, efectúe una doble pulsación en la cola.
    - c) 

Consulte las propiedades y edítelas según sea necesario. Si el recuadro de texto de propiedad está inhabilitado, la propiedad es de sólo lectura o solo se puede editar desde la línea de mandatos. Para obtener información sobre una propiedad, consulte la información de la propiedad en [Propiedades de cola](#) en la documentación de MQ Explorer.
  - Para ver y editar registros de autorizaciones para una cola:
    - a) Seleccione la cola en el widget.
    - b) Pulse  ... > **Gestionar registros de autorización**.
 

Los registros de autorizaciones muestran los permisos que los usuarios y administradores tienen sobre la cola seleccionada.

## Trabajar con temas

Puede utilizar el widget de temas en IBM MQ Console para añadir y suprimir temas y para ver y establecer las propiedades de un tema.

### Antes de empezar



Debe crear un widget de temas antes de poder utilizarlo. Para obtener más información sobre cómo crear widgets de objeto de IBM MQ, consulte [“Trabajar con objetos de IBM MQ”](#) en la página 100.






### Acerca de esta tarea

El widget de temas enumera los temas que existen para un gestor de colas específico. Puede seleccionar temas individuales de la lista con los que trabajar.

 No puede ver ni editar registros de autorizaciones de un tema en z/OS.

### Procedimiento

- Para añadir un tema:
  - a) Pulse el icono crear  en la barra de herramientas del widget de temas.
  - b) Escriba un nombre para el nuevo tema. Los caracteres válidos son letras y números y los caracteres ":", "/", "\_", y "%".
  - c) Especifique la serie de tema en la que publica mensajes para el tema. Para obtener más información, consulte [Propiedades de tema](#).
  - d) Pulse **Crear**. Se crea el nuevo tema.
- Para suprimir un tema:
  - a) Seleccione el tema que desea suprimir de la lista en el widget de temas.
  - b) Pulse el icono Suprimir  en la barra de herramientas del widget de temas.
  - c) Confirme que desea suprimir el tema pulsando **Suprimir**. Se suprime el tema.

- Para ver y editar las propiedades de un tema:
  - a) Seleccione el tema en el widget de temas.
  - b) Pulse el icono Propiedades  en la barra de herramientas del widget de temas. O bien, efectúe una doble pulsación en el tema.
  - c)  Consulte las propiedades y edítelas según sea necesario. Si el recuadro de texto de propiedad está inhabilitado, la propiedad es de sólo lectura o solo se puede editar desde la línea de mandatos. Para obtener información sobre una propiedad, consulte la información de la propiedad en [Propiedades de tema](#) en la MQ Explorer.
- Para publicar un mensaje en un tema:
  - a) Pulse el icono Transferir mensaje  en la barra de herramientas del widget de temas.
  - b) Escriba un mensaje en el campo **Mensaje**.
  - c) Especifique la serie de tema en la que publicar el mensaje en el campo **Serie de tema**.
  - d) Pulse **Publicar**.
- Para suscribirse a un tema:
  - a) Pulse el icono de suscripción  en la barra de herramientas del widget de temas.
  - b) Escriba la serie de tema a la que desea suscribirse en el campo **Serie de tema**.
  - c) Pulse **Suscribir**.
- Para ver y editar registros de autorizaciones para un tema:
  - a) Seleccione el tema en el widget de temas.
  - b) Pulse  ... > **Gestionar registros de autorización**.  
Los registros de autorizaciones muestran los permisos que los usuarios y administradores tienen sobre el tema seleccionado.

### **Trabajar con escuchas**


Puede utilizar el widget de escuchas en IBM MQ Console para añadir y suprimir escuchas, iniciar y detener escuchas, ver y establecer propiedades del escucha y gestionar los registros de autorizaciones de un escucha.

#### **Antes de empezar**


Debe crear un widget de escuchas antes de poder utilizarlo. Para obtener más información sobre cómo crear widgets de objeto de IBM MQ, consulte [“Trabajar con objetos de IBM MQ” en la página 100](#).





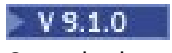
#### **Acerca de esta tarea**


El widget de escuchas enumera los escuchas que existen para un gestor de colas específico. Puede seleccionar escuchas individuales de la lista con los que trabajar.

 No puede utilizar el widget de escuchas en z/OS.

#### **Procedimiento**

- Para añadir un escucha TCP/IP:
  - a) Pulse el icono crear  en la barra de herramientas del widget de escuchas.

- b) Escriba un nombre para el escucha. Los caracteres válidos son letras y números y los caracteres ".", "/", "\_" y "%".
- c) Especifique un puerto TCP/IP disponible para el escucha. El número de puerto no debe exceder 65535.
- d) Pulse **Crear**. Se crea el nuevo escucha.
- Para suprimir un escucha:
  - a) Seleccione el escucha que desea suprimir de la lista del widget de escuchas.
  - b) Pulse el icono Suprimir  en la barra de herramientas del widget de escuchas.
  - c) Confirme que desea suprimir el escucha pulsando **Suprimir**. Se suprime el escucha.
- Para iniciar un escucha:
  - a) Seleccione el escucha que desea iniciar de la lista del widget de escuchas.
  - b) Pulse el icono de inicio  en la barra de herramientas del widget de escuchas.
- Para detener un escucha:
  - a) Seleccione el escucha que desea detener de la lista del widget de escuchas.
  - b) Pulse el icono de detención  en la barra de herramientas del widget de escuchas.
  - c) Confirme que desea detener el escucha pulsando **Detener**.
- Para ver y editar las propiedades de un escucha:
  - a) Seleccione el escucha en el widget de escuchas.
  - b) Pulse el icono de propiedades  en la barra de herramientas del widget de escuchas. O bien, efectúe una doble pulsación en el escucha.
  - c) 

Consulte las propiedades y edítelas según sea necesario. Si el recuadro de texto de propiedad está inhabilitado, la propiedad es de sólo lectura o solo se puede editar desde la línea de mandatos. Para obtener información sobre una propiedad, consulte la información de la propiedad en [Propiedades de escucha](#) en la MQ Explorer.
- Para ver y editar registros de autorizaciones para una escucha:
  - a) Seleccione el escucha en el widget de escuchas.
  - b) Pulse  ... > **Gestionar registros de autorización**. Los registros de autorizaciones muestran los permisos que los usuarios y administradores tienen sobre el escucha seleccionado.

### **Trabajar con canales**

Puede utilizar el widget de canales en la IBM MQ Console para añadir y suprimir canales, iniciar y detener canales, restablecer y resolver canales o hacer ping de canales. También puede ver y establecer las propiedades de un canal y gestionar los registros de autorizaciones del canal.

### **Antes de empezar**

Debe crear un widget de canales antes de poder utilizarlo. Para obtener más información sobre cómo crear widgets de objeto de IBM MQ, consulte ["Trabajar con objetos de IBM MQ"](#) en la página 100.








### **Acerca de esta tarea**




El widget de canales enumera los canales que existen para un gestor de colas específico. Puede seleccionar canales individuales de la lista para trabajar con ellos.

 No puede ver ni editar registros de autorizaciones de un canal en z/OS.



## Procedimiento

- Para añadir un canal:
  - a) Pulse el icono crear  en la barra de herramientas del widget de canales.
  - b) Escriba un nombre para el canal. Los caracteres válidos son letras y números y los caracteres " ", "/", "\_" y "%".
  - c) Seleccione el tipo de canal que desea añadir.
  - d) Si crea un canal emisor, un canal remitente de clúster o un canal peticionario, especifique el nombre de conexión. El nombre de conexión es el nombre del sistema que aloja el gestor de colas de destino. El formato del nombre es *nombre\_sistema(número\_puerto)*. *nombre\_sistema* es el nombre o la dirección IP del sistema que aloja el gestor de colas de destino y *número\_puerto* es el puerto que está utilizando el proceso de escucha del gestor de colas de destino.
  - e) Si crea un canal emisor o un canal servidor, especifique la cola de transmisión que corresponde al gestor de colas en el extremo receptor del canal.
  - f) Pulse **Crear**. Se crea el nuevo canal.
- Para suprimir un canal:
  - a) Seleccione el canal que desea suprimir de la lista del widget de canales.
  - b) Pulse el icono de supresión  en la barra de herramientas del widget.
  - c) Confirme que desea suprimir el canal pulsando **Suprimir**. Se suprime el canal.
- Para iniciar un canal:
  - a) Seleccione el canal que desea iniciar de la lista del widget de canales.
  - b) Pulse el icono de inicio  en la barra de herramientas del widget del gestor de colas.
- Para detener un canal:
  - a) Seleccione el canal que desea detener de la lista del widget de canales.
  - b) Pulse el icono de parada  en la barra de herramientas del widget del gestor de colas.
  - c) Confirme que desea detener el canal pulsando **Detener**.
- Para ver las propiedades de una cola:
  - a) Seleccione el canal en el widget de canales.
  - b) Pulse el icono de propiedades  en la barra de herramientas del widget de canales. O bien, efectúe una doble pulsación en el canal.
  - c)   
Consulte las propiedades y edítelas según sea necesario. Si el recuadro de texto de propiedad está inhabilitado, la propiedad es de sólo lectura o solo se puede editar desde la línea de mandatos. Para obtener información sobre una propiedad, consulte la información de propiedad en [Propiedades de canal](#) en la documentación de MQ Explorer.
- Para restablecer un canal:
  - a) Seleccione el canal en el widget de canales.
  - b) Pulse  > **Restablecer**.
  - c) Especifique un número de secuencia de mensaje.  
Debe restablecer un canal si no se inicia porque los dos extremos discrepan sobre el número de secuencia del siguiente mensaje a enviar. El número de secuencia de mensaje especifica dicho número.

- d) Pulse **Restablecer canal**.
- Para resolver un canal:
  - a) Seleccione el canal en el widget de canales.
  - b) Pulse  ... > **Resolver**.
  - c) Elija si desea confirmar o restituir el lote actual de mensajes pulsando **Confirmar** o **Restituir**.
- Para hacer ping de un canal:
  - a) Seleccione el canal en el widget de canales.
  - b) Pulse  ... > **Ping**.
- Para ver o editar registros de autorizaciones para un canal:
  - a) Seleccione el canal en el widget.
  - b) Pulse  ... > **Gestionar registros de autorización**.  
 Los registros de autorizaciones muestran los permisos que los usuarios y administradores tienen sobre el canal seleccionado.

### **Trabajar con canales de conexión cliente**

Puede utilizar el widget de canales de conexión cliente en IBM MQ Console para añadir y suprimir canales de conexión cliente en un gestor de colas, ver y establecer las propiedades y gestionar los registros de autorizaciones para el canal.

#### **Antes de empezar**





Debe crear un widget de canales de conexión cliente antes de poder utilizarlo. Para obtener más información sobre cómo crear widgets de objeto de IBM MQ, consulte [“Trabajar con objetos de IBM MQ”](#) en la página 100.

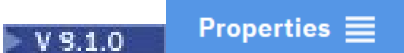


#### **Acerca de esta tarea**

El widget de canales de conexión cliente enumera los canales de conexión cliente que existen para un gestor de colas específico. Puede seleccionar canales de conexión cliente individuales de la lista para trabajar con ellos.

 No puede ver ni editar registros de autorizaciones de un canal de conexión cliente en z/OS.

#### **Procedimiento**

- Para añadir un canal de conexión cliente:
  - a) Pulse el icono crear  **Create**  en la barra de herramientas del widget de canales de conexión de cliente.
  - b) Escriba un nombre de canal de conexión cliente. Los caracteres válidos son letras y números y los caracteres ".", "/", "\_", y "%".
  - c) Especifique el nombre de conexión. El nombre de conexión es el nombre del sistema que aloja el gestor de colas de destino. El formato es *nombre\_sistema(número\_puerto)*, donde *nombre\_sistema* es el nombre o la dirección IP del sistema que aloja el gestor de colas de destino y *número\_puerto* es el puerto que el escucha del gestor de colas de destino está utilizando.
  - d) Pulse **Crear**. Se crea el nuevo canal de conexión cliente.
- Para suprimir un canal de conexión cliente.
  - a) Seleccione el canal de conexión cliente que desea suprimir de la lista en el widget de canales de conexión cliente.
  - b) Pulse el icono de supresión  **Delete**  en la barra de herramientas del widget.

- c) Confirme que desea suprimir el canal de conexión cliente pulsando **Suprimir**. El canal de conexión cliente se suprime.
- Para ver y editar las propiedades de un canal de conexión cliente:
  - a) Seleccione el canal de conexión cliente del widget de canales de conexión cliente.
  - b) Pulse el icono de propiedades  en la barra de herramientas del widget de canales de conexión de cliente. O bien, efectúe una doble pulsación en el canal de conexión cliente.
  - c)  Consulte las propiedades y edítelas según sea necesario. Si el recuadro de texto de propiedad está inhabilitado, la propiedad es de sólo lectura o solo se puede editar desde la línea de mandatos. Para obtener información sobre una propiedad, consulte la información de propiedad en [Propiedades de canal](#) en la documentación de MQ Explorer.
- Para ver y editar registros de autorizaciones para un canal de conexión cliente:
  - a) Seleccione el canal de conexión cliente del widget de canales de conexión cliente.
  - b) Pulse  ... > **Gestionar registros de autorización**. Los registros de autorizaciones muestran los permisos que los usuarios y administradores tienen sobre el canal de conexión cliente seleccionado.

### **Trabajar con información de autenticación**

Puede utilizar el widget de información de autenticación en IBM MQ Console para añadir y suprimir objetos de información de autenticación en un gestor de colas. También puede ver y establecer las propiedades y gestionar los registros de autorizaciones del canal.

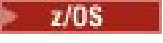
### **Antes de empezar**

Debe crear un widget de información de autenticación antes de poder utilizarlo. Para obtener más información sobre cómo crear widgets de objeto de IBM MQ, consulte [“Trabajar con objetos de IBM MQ”](#) en la página 100.


### **Acerca de esta tarea**




El widget de información de autenticación enumera la información de autenticación que existe para un gestor de colas específico. Puede seleccionar información de autenticación individual de la lista para trabajar con ella.

La información de autenticación del gestor de colas forma parte del soporte de IBM MQ para la seguridad de la capa de transporte (TLS). Estos objetos contienen las definiciones necesarias para realizar la comprobación de revocación de certificados mediante OCSP o listas de revocación de certificados (CRL) en los servidores LDAP y las definiciones necesarias para habilitar la comprobación de ID de usuario y contraseña.

 No puede utilizar IDPW LDAP ni ver ni editar los registros de autorizaciones de un objeto de información de autenticación en z/OS.

### **Procedimiento**

- Para añadir un objeto de información de autenticación:
  - a) Pulse el icono Crear  en la barra de herramientas del widget de información de autenticación.
  - b) Especifique el nombre del objeto de información de autenticación. Los caracteres válidos son letras y números y los caracteres ".", "/", "\_" y "%".
  - c) Especifique el tipo de objeto de información de autenticación.

- d) Especifique información adicional adecuada para el tipo de objeto:
- Para **CRL LDAP**, especifique el **Nombre del servidor LDAP**. Se trata del nombre de host, la dirección IPv4 en notación decimal con puntos o IPv6 en notación hexadecimal del host en el que se está ejecutando el servidor LDAP, con un número de puerto opcional.
  - Para **OCSP**, especifique el **URL del programa de respuesta OCSP**. Este URL es el URL del programa de respuesta OCSP (respondedor OCSP) que se utiliza para comprobar la revocación de certificados. Este valor debe ser un URL de HTTP que contenga el nombre de host y el número de puerto del respondedor OCSP. Si el programa de respuesta OCSP está utilizando el puerto 80, que es el valor predeterminado para HTTP, entonces el número de puerto se puede omitir. Los URL de HTTP se definen en la RFC 1738.
  - Para **IDPW OS**, no hay requisitos adicionales.
  - Para **IDPW LDAP**, especifique el **nombre de servidor LDAP** y el nombre **Usuario abreviado**. El nombre del servidor LDAP es el nombre de host, la dirección IPv4 en notación decimal con puntos o IPv6 en notación hexadecimal del host en el que se está ejecutando el servidor LDAP, con un número de puerto opcional. El nombre de usuario abreviado es el campo del registro de usuario de LDAP que se utiliza como nombre abreviado para la conexión.
- e) Pulse **Crear**.
- Para suprimir un objeto de información de autenticación:
    - a) Seleccione el objeto de información de autenticación que desea suprimir en la lista del widget.
    - b) Pulse el icono de supresión  en la barra de herramientas del widget.
    - c) Confirme que desea suprimir el objeto de información de autenticación pulsando **Suprimir**. Se suprime el objeto.
  - Para ver y editar las propiedades de un objeto de información de autenticación:
    - a) Seleccione el objeto de información de autenticación en el widget.
    - b) Pulse el icono de propiedades  en la barra de herramientas del widget. O bien, efectúe una doble pulsación en el objeto de información de autenticación.
    - c) Consulte las propiedades y edítelas según sea necesario. Si el recuadro de texto de propiedad está inhabilitado, la propiedad es de sólo lectura o solo se puede editar desde la línea de mandatos.
  - Para ver y editar los registros de autorizaciones de un objeto de información de autenticación:
    - a) Seleccione el objeto de información de autenticación en el widget de información de autenticación.
    - b) Pulse  **> Gestionar registros de autorización**. Los registros de autorizaciones muestran los permisos que los usuarios y los administradores tienen sobre el objeto de información de autenticación seleccionado.

## **Trabajar con suscripciones**

Puede utilizar el widget de suscripciones en IBM MQ Console para añadir y suprimir suscripciones en un gestor de colas, ver y establecer las propiedades y gestionar los registros de autorizaciones para las suscripciones.

### **Antes de empezar**

Debe crear un widget de suscripciones antes de poder utilizarlo. Para obtener más información sobre cómo crear widgets de objeto de IBM MQ, consulte [“Trabajar con objetos de IBM MQ”](#) en la página 100.

### **Acerca de esta tarea**


Las suscripciones se emiten para un gestor de colas y contienen información sobre las publicaciones que el suscriptor desea recibir:

- La serie de tema en la que está interesado el suscriptor; este tema se puede resolver en varias series de tema si se utilizan comodines.
- Se debe aplicar una serie de selección opcional para los mensajes publicados.
- El nombre de la cola en la que se colocan las publicaciones seleccionadas.


Para obtener más información sobre las suscripciones, consulte [Suscriptores y suscripciones](#) y [DEFINE SUB](#).

## Procedimiento


- Para añadir un objeto de suscripción:

- Pulse el icono Crear  en la barra de herramientas del widget de suscripciones.
- Especifique el nombre del objeto. Los caracteres válidos son letras y números y los caracteres ".", "/", "\_" y "%".
- Seleccione una **Clase de destino** de **Gestionado** o **Proporcionado**. Si selecciona **Gestionado**, se crea un destino en el gestor de colas local.
- Si selecciona una clase de destino de **Proporcionado**, en el campo **Destino**, especifique el nombre de la cola a la que se reenviarán los mensajes para esta suscripción.
- En el campo **Serie de tema**, especifique la serie de tema a la que desea suscribirse.
- Seleccione un valor **Uso de comodín**. Seleccione **Comodín a nivel de carácter** si desea especificar que los caracteres comodín representan partes de series. Seleccione **Comodín a nivel de tema** si desea especificar que los caracteres comodín representan partes de la jerarquía de temas.
- Seleccione un **Ámbito**. Seleccione **Todos** para reenviar la suscripción a todos los gestores de colas conectados directamente mediante un colectivo o una jerarquía de publicación/suscripción. Seleccione **Gestor de colas** para que la suscripción reenvíe los mensajes que se publican sobre el tema solo dentro de este gestor de colas.
- Opcional: Especifique un **Selector**. Una serie de selección es una expresión que se aplica a una publicación para determinar si coincide con una suscripción.
- Pulse **Crear**.

- Para suprimir un objeto de suscripción:

- Seleccione el objeto de suscripción que desea suprimir de la lista en el widget de suscripciones.
- Pulse el icono de supresión  en la barra de herramientas del widget.
- Confirme que desea suprimir el objeto de suscripción pulsando **Suprimir**. Se suprime el objeto.

- Para ver y editar las propiedades de un objeto de suscripción:

- Seleccione el objeto de suscripción en el widget.
- Pulse el icono de propiedades  en la barra de herramientas del widget. O bien, efectúe una doble pulsación en el objeto de suscripción.
- Consulte las propiedades y edítelas según sea necesario. Si el recuadro de texto de propiedad está inhabilitado, la propiedad es de sólo lectura o solo se puede editar desde la línea de mandatos.

### **Trabajar con registros de autenticación de canal**

Puede utilizar el widget de registro de autenticación de canal en la IBM MQ Console para añadir y suprimir registros de autenticación de canal en un gestor de colas. También puede ver y establecer las propiedades de los registros de autenticación de canal.

## Antes de empezar



Debe crear un widget de registro de autenticación de canal antes de poder utilizarlo. Para obtener más información sobre cómo crear widgets de objeto de IBM MQ, consulte [“Trabajar con objetos de IBM MQ”](#) en la página 100.

## Acerca de esta tarea

Para ejercer un control más preciso sobre el acceso que se otorga a los sistemas de conexión en un nivel de canal, puede utilizar registros de autenticación de canal.

Para garantizar la seguridad, puede utilizar registros de autenticación de canal de bloqueo para bloquear el acceso a los canales. También puede utilizar registros de autenticación de canal de correlación de direcciones para permitir el acceso a usuarios especificados. Para obtener más información sobre los registros de autenticación de canal, consulte [Registros de autenticación de canal](#).

## Procedimiento

- Para añadir un registro de autenticación de canal con una identidad de nombre distinguido SSL/TLS, consulte [“Creación de registros de autenticación de canal con una identidad de nombre distinguido SSL/TLS”](#) en la página 110.
- Para añadir un registro de autenticación de canal con una identidad de ID de usuario de aplicación cliente, consulte [“Creación de registros de autenticación de canal con una identidad de ID de usuario de aplicación cliente”](#) en la página 112.
- Para añadir un registro de autenticación de canal con una identidad de nombre de gestor de colas remoto, consulte [“Creación de registros de autenticación de canal con una identidad de nombre de gestor de colas remoto”](#) en la página 113.
- Para añadir un registro de autenticación de canal con una identidad de dirección, consulte [“Creación de registros de autenticación de canal con una identidad de dirección IP”](#) en la página 114.
- Para suprimir un registro de autenticación de canal:
  - a) Seleccione el registro de autenticación de canal que desea suprimir de la lista en el widget de registros de autenticación de canal.
  - b) Pulse el icono de supresión  en la barra de herramientas del widget.
  - c) Confirme que desea suprimir el registro de autenticación de canal pulsando **Suprimir**. El registro de autenticación de canal se suprime.
- Para ver y editar las propiedades de un registro de autenticación de canal:
  - a) Seleccione el registro de autenticación de canal que desea editar de la lista en el widget de registro de autenticación de canal.
  - b) Pulse el icono de propiedades  en la barra de herramientas del widget. O bien, efectúe una doble pulsación en el registro de autenticación de canal.
  - c) Consulte las propiedades y edítelas según sea necesario. Si el recuadro de texto de propiedad está inhabilitado, la propiedad es de sólo lectura o solo se puede editar desde la línea de mandatos.

### *Creación de registros de autenticación de canal con una identidad de nombre distinguido SSL/TLS*


Puede utilizar el widget de registros de autenticación de canal para crear registros de autenticación de canal de permiso, bloqueo y aviso con una identidad de nombre distinguido SSL/TLS. La identidad de nombre distinguido SSL/TLS coincide con los usuarios que presentan un certificado personal SSL o TLS que contiene un nombre distinguido especificado.

## Antes de empezar

Debe crear un widget de registro de autenticación de canal antes de poder utilizarlo. Para obtener más información sobre cómo crear widgets de objeto de IBM MQ, consulte [“Trabajar con objetos de IBM MQ”](#) en la [página 100](#).

## Procedimiento

- Para añadir un registro de autenticación de canal:

- Pulse el icono crear  en la barra de herramientas del widget de registro de autenticación de canal.
- Seleccione el **Tipo de regla** para indicar qué tipo de regla desea en el registro de autenticación de canal:
  - Seleccione **Permitir** para permitir el acceso a las conexiones de entrada.
  - Seleccione **Bloquear** para bloquear el acceso a las conexiones de entrada.
  - Seleccione **Avisos** para avisar sobre el acceso a las conexiones de entrada que se bloquearían. Se permite el acceso a la conexión y se informa de un mensaje de error. Si están configurados los sucesos, se crea un mensaje de suceso que muestra los detalles de lo que se bloquearía. Solo se notifican las reglas coincidentes.
- Seleccione el tipo de identidad **Nombre distinguido SSL/TLS** de la lista.
- Pulse **Siguiente**
- Especifique un **Perfil de canal**.

El perfil de canal es el nombre del canal o conjunto de canales para el que está estableciendo la autenticación de canal. El perfil puede contener comodines para que pueda bloquear una serie de canales. Por ejemplo, el perfil `a1phadelta*` bloquea canales denominados `alphadelta1`, `alphadelta2`, `alphadelta3`, etc.
- Especifique el **Nombre de igual**. Por ejemplo, `CN=John Smith, O=IBM, OU=Test, C=GB`. Para obtener más información sobre los nombres de iguales, consulte [Reglas de WebSphere MQ para los valores de SSLPEER](#).
- Opcional: Especifique el filtro **Dirección** que se utiliza. La dirección es la dirección IP que se espera en el otro extremo del canal.
- Opcional: Especifique el **Nombre de ID de usuario Cert de SSL**. El nombre de ID de usuario Cert de SSL es el nombre de la entidad emisora de certificados mediante la que se debe emitir el certificado SSL/TLS.
- Opcional: Pulse **Siguiente**.
- Opcional: En el caso de un tipo de regla **Permitir**, puede especificar opcionalmente el **Origen de usuario** para el registro de autenticación de canal. El origen de usuario especifica el origen del ID de usuario que se utiliza cuando la conexión de entrada coincide con el nombre distinguido SSL/TLS.
  - La opción **Canal** especifica que las conexiones de entrada que coinciden con la correlación utilizan el ID de usuario de flujo o cualquier usuario que esté definido en el objeto de canal.
  - La opción **Correlación** especifica que las conexiones de entrada que coinciden con la correlación utilizan el ID de usuario que se especifica en el campo **ID de usuario de MCA**.
- Opcional: Pulse **Siguiente**.
- Opcional: Especifique una **Descripción** para el registro de autenticación de canal.
- Pulse **Crear**. Se crea el nuevo registro de autenticación de canal.



## Creación de registros de autenticación de canal con una identidad de ID de usuario de aplicación cliente

Puede utilizar el widget de registros de autenticación de canal para crear registros de autenticación de canal de permiso, bloqueo y aviso con una identidad de ID de usuario de aplicación cliente. La identidad de ID de usuario de aplicación cliente coincide con los ID de aplicación cliente de un canal de conexión cliente.

### Antes de empezar

Debe crear un widget de registro de autenticación de canal antes de poder utilizarlo. Para obtener más información sobre cómo crear widgets de objeto de IBM MQ, consulte [“Trabajar con objetos de IBM MQ” en la página 100.](#)

### Procedimiento

- Para añadir un registro de autenticación de canal:
  - a) Pulse el icono crear   en la barra de herramientas del widget de registro de autenticación de canal.
  - b) Seleccione el **Tipo de regla** para indicar qué tipo de regla desea en el registro de autenticación de canal:
    - Seleccione **Permitir** para permitir el acceso a las conexiones de entrada.
    - Seleccione **Bloquear** para bloquear el acceso a las conexiones de entrada.
    - Seleccione **Avisos** para avisar sobre el acceso a las conexiones de entrada que se bloquearían. Se permite el acceso a la conexión y se informa de un mensaje de error. Si están configurados los sucesos, se crea un mensaje de suceso que muestra los detalles de lo que se bloquearía. Solo se notifican las reglas coincidentes.
  - c) Seleccione el tipo de identidad **ID de usuario de aplicación cliente** de la lista.
  - d) Pulse **Siguiente**.
  - e) Especifique un **Perfil de canal**.

El nombre del perfil de canal es el nombre del canal o conjunto de canales para el que está estableciendo la autenticación de canal. El perfil puede contener comodines para que pueda bloquear una serie de canales. Por ejemplo, el perfil `a1phadelta*` bloquea canales denominados `alphadelta1`, `alphadelta2`, `alphadelta3`, etc.
  - f) Especifique el **ID de usuario cliente**. El ID de usuario cliente es el ID de usuario del cliente que desea permitir, bloquear o sobre el que desea advertir.
  - g) Opcional: Especifique el filtro **Dirección** que se utiliza. La dirección es la dirección IP que se espera en el otro extremo del canal.
  - h) Opcional: Pulse **Siguiente**.
  - i) Opcional: En el caso de un tipo de regla **Permitir**, puede especificar opcionalmente el **Origen de usuario** para el registro de autenticación de canal. El origen de usuario especifica el origen del ID de usuario que se utiliza cuando la conexión de entrada coincide con el ID de usuario cliente.
    - La opción **Canal** especifica que las conexiones de entrada que coinciden con la correlación utilizan el ID de usuario de flujo o cualquier usuario que esté definido en el objeto de canal.
    - La opción **Correlación** especifica que las conexiones de entrada que coinciden con la correlación utilizan el ID de usuario que se especifica en el campo **ID de usuario de MCA**.
  - j) Opcional: Pulse **Siguiente**.
  - k) Opcional: Especifique una **Descripción** para el registro de autenticación de canal.
  - l) Pulse **Crear**. Se crea el nuevo registro de autenticación de canal.





## Creación de registros de autenticación de canal con una identidad de nombre de gestor de colas remoto

Puede utilizar el widget de registros de autenticación de canal para crear registros de autenticación de canal de permiso, bloqueo y aviso con una identidad de nombre de gestor de colas remoto. La identidad de nombre de gestor de colas remoto coincide con el gestor de colas especificado.

### Antes de empezar

Debe crear un widget de registro de autenticación de canal antes de poder utilizarlo. Para obtener más información sobre cómo crear widgets de objeto de IBM MQ, consulte [“Trabajar con objetos de IBM MQ” en la página 100.](#)

### Procedimiento

- Para añadir un registro de autenticación de canal:
  - a) Pulse el icono crear   en la barra de herramientas del widget de registro de autenticación de canal.
  - b) Seleccione el **Tipo de regla** para indicar qué tipo de regla desea en el registro de autenticación de canal:
    - Seleccione **Permitir** para permitir el acceso a las conexiones de entrada.
    - Seleccione **Bloquear** para bloquear el acceso a las conexiones de entrada.
    - Seleccione **Avisos** para avisar sobre el acceso a las conexiones de entrada que se bloquearían. Se permite el acceso a la conexión y se informa de un mensaje de error. Si están configurados los sucesos, se crea un mensaje de suceso que muestra los detalles de lo que se bloquearía. Solo se notifican las reglas coincidentes.
  - c) Seleccione el tipo de identidad **Nombre de gestor de colas remoto** de la lista.
  - d) Pulse **Siguiente**
  - e) Especifique un **Nombre de perfil**.

El nombre de perfil es el nombre del canal o conjunto de canales para el que está estableciendo la autenticación de canal. El perfil puede contener comodines para que pueda bloquear una serie de canales. Por ejemplo, el perfil `a1phade1ta*` bloquea canales denominados `alphadelta1`, `alphadelta2`, `alphadelta3`, etc.
  - f) Especifique el **Nombre del gestor de colas**. El nombre del gestor de colas especifica el nombre del gestor de colas remoto que desea permitir, bloquear o advertir acerca de él.
  - g) Opcional: Especifique el filtro **Dirección** que se utiliza. La dirección es la dirección IP que se espera en el otro extremo del canal.
  - h) Opcional: Pulse **Siguiente**.
  - i) Opcional: En el caso de un tipo de regla **Permitir**, puede especificar opcionalmente el **Origen de usuario** para el registro de autenticación de canal. El origen de usuario especifica el origen del ID de usuario que se utiliza cuando la conexión de entrada coincide con el nombre de gestor de colas remoto.
    - La opción **Canal** especifica que las conexiones de entrada que coinciden con la correlación utilizan el ID de usuario de flujo o cualquier usuario que esté definido en el objeto de canal.
    - La opción **Correlación** especifica que las conexiones de entrada que coinciden con la correlación utilizan el ID de usuario que se especifica en el campo **ID de usuario de MCA**.
  - j) Opcional: Pulse **Siguiente**.
  - k) Opcional: Especifique una **Descripción** para el registro de autenticación de canal.
  - l) Pulse **Crear**. Se crea el nuevo registro de autenticación de canal.

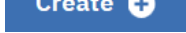
Puede utilizar el widget de registros de autenticación de canal para crear registros de autenticación de canal de permiso, bloqueo y aviso con una identidad de dirección. La identidad de dirección coincide con direcciones IP específicas.

## Antes de empezar

Debe crear un widget de registro de autenticación de canal antes de poder utilizarlo. Para obtener más información sobre cómo crear widgets de objeto de IBM MQ, consulte [“Trabajar con objetos de IBM MQ” en la página 100.](#)

## Procedimiento

- Para añadir un registro de autenticación de canal:

- a) Pulse el icono crear  en la barra de herramientas del widget de registro de autenticación de canal.
- b) Seleccione el **Tipo de regla** para indicar qué tipo de regla desea en el registro de autenticación de canal:
  - Seleccione **Permitir** para permitir el acceso a las conexiones de entrada.
  - Seleccione **Bloquear** para bloquear el acceso a las conexiones de entrada.
  - Seleccione **Avisos** para avisar sobre el acceso a las conexiones de entrada que se bloquearían. Se permite el acceso a la conexión y se informa de un mensaje de error. Si están configurados los sucesos, se crea un mensaje de suceso que muestra los detalles de lo que se bloquearía. Solo se notifican las reglas coincidentes.
- c) Seleccione el tipo de identidad **Dirección** de la lista.
- d) Pulse **Siguiente**
- e) Opcional: Para un tipo de regla **Bloquear** o **Avisos**, especifique **Cuándo efectuar la coincidencia**. Puede seleccionar una de estas opciones:
  - **En el escucha**. Esta opción intenta hacer coincidir la regla en el escucha.
  - **En el canal**. Esta opción intenta hacer coincidir la regla en el canal.
- f) Especifique un **Nombre de perfil**.  
El nombre de perfil es el nombre del canal o conjunto de canales para el que está estableciendo la autenticación de canal. El perfil puede contener comodines para que pueda bloquear una serie de canales. Por ejemplo, el perfil `a1phade1ta*` bloquea canales denominados `alphadelta1`, `alphadelta2`, `alphadelta3`, etc.
- g) Especifique una **Dirección**. Se trata de la dirección IP o de una lista separada por comas de direcciones IP que están permitidas o bloqueadas.
- h) Opcional: Pulse **Siguiente**.
- i) Opcional: En el caso de un tipo de regla **Permitir**, puede especificar opcionalmente el **Origen de usuario** para el registro de autenticación de canal. El origen de usuario especifica el origen del ID de usuario que se utiliza cuando la conexión de entrada coincide con el nombre de gestor de colas remoto.
  - La opción **Canal** especifica que las conexiones de entrada que coinciden con la correlación utilizan el ID de usuario de flujo o cualquier usuario que esté definido en el objeto de canal.
  - La opción **Correlación** especifica que las conexiones de entrada que coinciden con la correlación utilizan el ID de usuario que se especifica en el campo **ID de usuario de MCA**.
- j) Opcional: Pulse **Siguiente**.
- k) Opcional: Especifique una **Descripción** para el registro de autenticación de canal.
- l) Pulse **Crear**. Se crea el nuevo registro de autenticación de canal.



## Creación de registros de autenticación de canal con una identidad de ID de usuario asignado final

Puede utilizar el widget de registros de autenticación de canal para crear registros de autenticación de canal de bloqueo y aviso con una identidad de ID de usuario asignado final. La identidad de ID de usuario asignado final coincide con la lista de los ID de usuario especificados de un canal servidor.

### Antes de empezar

Debe crear un widget de registro de autenticación de canal antes de poder utilizarlo. Para obtener más información sobre cómo crear widgets de objeto de IBM MQ, consulte [“Trabajar con objetos de IBM MQ” en la página 100.](#)

### Procedimiento

- Para añadir un registro de autenticación de canal:
  - a) Pulse el icono crear   en la barra de herramientas del widget de registro de autenticación de canal.
  - b) Seleccione el **Tipo de regla** para indicar qué tipo de regla desea en el registro de autenticación de canal:
    - Seleccione **Bloquear** para bloquear el acceso a las conexiones de entrada.
    - Seleccione **Avisos** para avisar sobre el acceso a las conexiones de entrada que se bloquearían. Se permite el acceso a la conexión y se informa de un mensaje de error. Si están configurados los sucesos, se crea un mensaje de suceso que muestra los detalles de lo que se bloquearía. Solo se notifican las reglas coincidentes.
  - c) Seleccione el tipo de identidad **ID de usuario asignado final** de la lista.
  - d) Pulse **Siguiente**
  - e) Especifique un **Nombre de perfil**.


El nombre de perfil es el nombre del canal o conjunto de canales para el que está estableciendo la autenticación de canal. El perfil puede contener comodines para que pueda bloquear una serie de canales. Por ejemplo, el perfil `alphadelta*` bloquea canales denominados `alphadelta1`, `alphadelta2`, `alphadelta3`, etc.
  - f) Especifique la **Lista de usuarios**. La lista de usuarios es una lista separada por comas de los ID de usuario que se van a bloquear desde el canal.
  - g) Opcional: Pulse **Siguiente**.
  - h) Opcional: Especifique una **Descripción** para el registro de autenticación de canal.
  - i) Pulse **Crear**. Se crea el nuevo registro de autenticación de canal.

## Trabajar con registros de autorizaciones


Puede controlar el acceso que los grupos tienen a gestores de colas y objetos de IBM MQ especificando un registro de autorizaciones para dicho grupo.

### Acerca de esta tarea

Puede ajustar el acceso que un grupo de usuarios de mensajería tiene a un determinado gestor de colas o a un objeto de IBM MQ utilizando los registros de autorizaciones. El registro de autorizaciones se configura de la misma manera para todos los tipos de objetos utilizando el mismo procedimiento, aunque los permisos reales que configure dependen del tipo de objeto.

Por ejemplo, compare los diferentes permisos disponibles para un gestor de colas y una cola, tal como se ilustra en las imágenes siguientes: 

## Authority records for 'qm3'

Delete  1 item selected [Cancel](#)

▲ Entity name	Entity type
mqm	Group
mqsystem	User

Total: 2 Last updated: 3:40:18 PM

**Administration**

- Change
- Delete
- Display
- Ctrl

**Context**

- Set all context
- Set identity context

**MQI**

- Alternate user authority
- Connect
- Inquire
- Set
- System

Check all


Uncheck all

Close

Save

V 9.1.0


## Authority records for 'q1' on qm3

Delete  1 item selected <a href="#">Cancel</a>	
▲ Entity name	Entity type
mqm	Group
mqsystem	User



<input checked="" type="checkbox"/> Administration	<input checked="" type="checkbox"/> Context	<input checked="" type="checkbox"/> MQI
<input checked="" type="checkbox"/> Change	<input checked="" type="checkbox"/> Pass all context	<input checked="" type="checkbox"/> Browse
<input checked="" type="checkbox"/> Clear	<input checked="" type="checkbox"/> Pass identity context	<input checked="" type="checkbox"/> Inquire
<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Set all context	<input checked="" type="checkbox"/> Get
<input checked="" type="checkbox"/> Display	<input checked="" type="checkbox"/> Set identity context	<input checked="" type="checkbox"/> Put
		<input checked="" type="checkbox"/> Set

[Check all](#) [Uncheck all](#) [Close](#) [Save](#)

 No puede trabajar con registros de autorizaciones en z/OS.

### Procedimiento

- Para ver o editar un registro de autorizaciones para un objeto de IBM MQ:
  - a) Seleccione el objeto en un widget del panel de instrumentos. El gestor de colas asociado debe estar en ejecución.
  - b) En la barra de herramientas del widget adecuado, seleccione ... > **Gestionar registros de autorización.**
  - c) Seleccione el grupo cuyo registro de autorizaciones desea ver. Se muestran las autorizaciones de ese grupo.
  - d) Seleccione o deseleccione las autorizaciones según sea necesario. Hay diferentes autorizaciones disponibles dependiendo del tipo de objeto para el que esté creando un registro de autorizaciones.
  - e) Pulse **Guardar.**
- Para ver o editar un registro de autorizaciones para crear un gestor de colas:
  - a) Seleccione el gestor de colas en un widget de gestor de colas del panel de instrumentos. El gestor de colas debe estar en ejecución.
  - b) En la barra de herramientas del widget, seleccione ... > **Gestionar registros de autorización de creación.**
  - c) Seleccione el grupo cuya creación del registro de autorizaciones desea ver. Se muestran las autorizaciones de ese grupo.

- d) Seleccione o deseleccione la creación de autorizaciones según sea necesario.
- e) Pulse **Guardar**.
- Para crear un registro de autorizaciones para un objeto de IBM MQ:
  - a) Seleccione el objeto de IBM MQ en un widget del panel de instrumentos. El gestor de colas asociado debe estar en ejecución.
  - b) En la barra de herramientas del widget, seleccione ... > **Gestionar registros de autorización**.
  - c) Pulse el icono de signo más .
  - d) Especifique el nombre del usuario o el grupo para el que está creando el registro de autorizaciones. El usuario o el grupo debe existir.
  - e) Seleccione el **Tipo de entidad** para especificar si la entidad es un usuario o un grupo.
  - f) Pulse **Crear**.
  - g) Seleccione o deseleccione las autorizaciones que desea que tenga el usuario o el grupo. Hay diferentes autorizaciones disponibles para cada tipo de objeto.
  - h) Pulse **Guardar**.
- Para crear un registro de autorizaciones para crear objetos en un gestor de colas:
  - a) Seleccione el gestor de colas en un widget del panel de instrumentos. El gestor de colas debe estar en ejecución.
  - b) En la barra de herramientas del widget, seleccione ... > **Gestionar registros de autorización de creación**.
  - c) Pulse el icono de creación .
  - d) Especifique el nombre del usuario o el grupo para el que está creando el registro de autorizaciones. El usuario o el grupo debe existir.
  - e) Seleccione el **Tipo de entidad** para especificar si la entidad es un usuario o un grupo.
  - f) Pulse **Crear**.
  - g) Seleccione o deseleccione las autorizaciones de creación que desea que tenga el usuario o el grupo.
  - h) Pulse **Guardar**.

## Supervisión del uso de recursos de sistema

Puede utilizar el widget de gráficos en IBM MQ Console para ver los datos de supervisión para los gestores de colas.


### Acerca de esta tarea

Puede añadir el widget de gráficos al panel de instrumentos y luego configurarlo para supervisar un aspecto determinado de uso del recurso. Puede crear muchas instancias del widget de gráficos para visualizar datos distintos. Los datos se muestran en un formato de diagrama.

Para poder iniciar el recurso del sistema de supervisión, el usuario o grupo debe formar parte de MQWebAdmin o MQWebAdminRO. Una vez que se ha iniciado la supervisión, cualquier MQWebUser autenticado puede acceder a la corriente de datos y realizar la supervisión añadiendo el widget de gráficos.

Los datos se recopilan a intervalos de 10 segundos. El eje X del gráfico muestra una línea temporal. El eje Y muestra las unidades correspondientes al recurso que está visualizando. El eje Y se redimensiona dinámicamente para dar cabida a los datos que se devuelven.


Debe tener como mínimo un gestor de colas en ejecución para poder configurar un widget de gráfico.

 No puede supervisar el uso de recursos del sistema en z/OS.

## Procedimiento


1. Añada un widget de gráficos al panel de instrumentos:

a) Pulse el icono Añadir widget



b) Seleccione **Gráficos**.

2. Configure el widget de gráficos para mostrar los datos:

a) Pulse el icono configurar  en la barra de título del widget de gráficos.

b) Opcional: Especifique un **Título de widget**. Este título se muestra en la barra de título del widget.

c) Seleccione la **Clase de recurso** que desea supervisar:

**Unidades centrales de proceso de la plataforma**

Supervise el uso de las CPU.

**Almacenes de datos persistentes de la plataforma**

Supervise el uso del recurso de disco.

**Estadísticas de uso de la API**

Supervise las llamadas de API.

**Estadísticas de uso por cola de la API**

Supervise las llamadas de API por colas individuales. Cuando selecciona esta clase, debe especificar el nombre de cola que se va a supervisar en el campo **Objeto**.

d) Seleccione el **Tipo de recurso** que desea supervisar.

Los tipos de recursos que están disponibles para seleccionarlos dependen de la clase de recurso que se seleccione. En la tabla siguiente se muestran los tipos de recursos:

Clase	Tipo	Descripción
Unidades centrales de proceso de la plataforma	Rendimiento de CPU - toda la plataforma	Seleccione este tipo para ver los datos de rendimiento de las CPU y la memoria.
	Rendimiento de CPU - ejecución del gestor de colas	Seleccione este tipo para ver los datos de rendimiento de las CPU y la memoria que están relacionados con los gestores de colas que está supervisando. Debe haber un gestor de colas en ejecución para que lo pueda supervisar. Si supervisa los resultados de más de un gestor de colas, se utilizan distintos colores para distinguir los datos de rendimiento del gráfico.
Almacenes de datos persistentes de la plataforma	Uso del disco - toda la plataforma	Seleccione este tipo para ver los datos de rendimiento del uso de disco global.

<i>Tabla 7. Tipos de recursos (continuación)</i>		
<b>Clase</b>	<b>Tipo</b>	<b>Descripción</b>
	Uso de disco - ejecución de gestores de colas	Seleccione este tipo para ver los datos de rendimiento para el uso de disco que está relacionado con los gestores de colas que está supervisando. Debe haber un gestor de colas en ejecución para que lo pueda supervisar. Si supervisa los resultados de más de un gestor de colas, se utilizan distintos colores para distinguir los datos de rendimiento del gráfico.
	Uso de disco - registro de recuperación del gestor de colas	Seleccione este tipo para ver los datos sobre cómo se está utilizando el almacenamiento de disco para el registro de recuperación de cada gestor de colas que está supervisando.
Estadísticas de uso de la API	MQCONN y MQDISC	Seleccione este tipo para ver datos sobre las llamadas MQCONN y MQDISC.
	MQOPEN y MQCLOSE	Seleccione este tipo para ver datos sobre las llamadas MQOPEN y MQCLOSE.
	MQINQ y MQSET	Seleccione este tipo para ver datos sobre las llamadas MQINQ y MQSET.
	MQPUT	Seleccione este tipo para ver datos sobre llamadas relacionadas con MQPUT.
	MQGET	Seleccione este tipo para ver datos sobre llamadas relacionadas con MQGET.
	Confirmar y retrotraer	Seleccione este tipo para ver información sobre el uso de los puntos de sincronización por parte del gestor de colas.
	Suscribir	Seleccione este tipo para ver datos que están relacionados con las llamadas MQSUB.
	Publicar	Seleccione este tipo para ver datos sobre mensajes publicados.
Estadísticas de uso por cola de la API	MQOPEN y MQCLOSE	Seleccione este tipo para ver datos sobre las llamadas MQOPEN y MQCLOSE para la cola especificada.
	MQINQ y MQSET	Seleccione este tipo para ver datos sobre las llamadas MQINQ y MQSET para la cola especificada.
	MQPUT y MQPUT1	Seleccione este tipo para ver datos sobre las llamadas relacionadas con MQPUT y relacionadas con MQPUT1 para la cola especificada.



<i>Tabla 7. Tipos de recursos (continuación)</i>		
<b>Clase</b>	<b>Tipo</b>	<b>Descripción</b>
	MQGET	Seleccione este tipo para ver datos sobre las llamadas relacionadas con MQGET y MQSET para la cola especificada.

e) Seleccione el **Elemento de recurso** que desea supervisar:

Los elementos de recurso que están disponibles para seleccionarlos dependen de la clase de recurso y del tipo de recurso que se seleccionen. En las tablas siguientes se muestran los elementos de recurso:

<i>Tabla 8. Elementos para recursos de unidades centrales de proceso de la plataforma</i>		
<b>Tipo</b>	<b>Elemento</b>	<b>Descripción</b>
Rendimiento de CPU - toda la plataforma	Porcentaje de tiempo de CPU del usuario	Muestra el porcentaje de CPU ocupada en el estado de usuario.
	Porcentaje de tiempo de CPU del sistema	Muestra el porcentaje de CPU ocupada en el estado del sistema.
	Carga de CPU - promedio de un minuto	Muestra el promedio de carga a lo largo de 1 minuto.
	Carga de CPU - promedio de cinco minutos	Muestra el promedio de carga a lo largo de 5 minutos.
	Carga de CPU - promedio de quince minutos	Muestra el promedio de carga a lo largo de quince minutos.
	Porcentaje libre de RAM	Muestra el porcentaje de memoria RAM libre.
	Total de bytes de RAM	Muestra el número total de bytes de RAM configurada.
Rendimiento de CPU - ejecución del gestor de colas	Tiempo de CPU de usuario – estimación de porcentaje para el gestor de colas	Calcula el porcentaje de uso de CPU en el estado de usuario para los procesos que están relacionados con los gestores de colas que se están supervisando.
	Tiempo de CPU del sistema – estimación de porcentaje para el gestor de colas	Calcula el porcentaje de uso de CPU en el estado del sistema para los procesos que están relacionados con los gestores de colas que se están supervisando.
	Total de bytes de RAM - cálculo de los gestores de colas	Calcula el total de bytes de RAM en uso por parte de los gestores de colas que se están supervisando.

<i>Tabla 9. Elementos para recursos de almacenes de datos persistentes de la plataforma</i>		
<b>Tipo</b>	<b>Elemento</b>	<b>Descripción</b>
Uso del disco - toda la plataforma	Sistema de archivos de rastreo MQ - bytes en uso	Muestra el número de bytes de almacenamiento en disco que el sistema de archivos de rastreo está utilizando.
	Sistema de archivos de rastreo MQ - espacio libre	Muestra el almacenamiento en disco que está reservado para el sistema de archivos de rastreo que está libre.
	Sistema de archivos de errores MQ - bytes en uso	Muestra el número de bytes de almacenamiento en disco que los datos de error utilizan.
	Sistema de archivos de errores MQ - espacio libre	Muestra el almacenamiento en disco reservado para los datos de error que está libre.
	Recuento de archivos de MQ FDC	Muestra el número actual de archivos FDC.
Uso de disco - ejecución de gestores de colas	Sistema de archivos del gestor de colas - bytes en uso	Muestra el número de bytes de almacenamiento de disco que los archivos del gestor de colas utilizan para los gestores de colas que está supervisando.
	Sistema de archivos del gestor de colas - espacio libre	Muestra el almacenamiento en disco reservado para los archivos del gestor de colas que está libre.
Uso de disco - registro de recuperación del gestor de colas	Registro - bytes en uso	Muestra el número de bytes de almacenamiento de disco que se utiliza para los registros de recuperación de los gestores de cola que está supervisando.
	Registro – máx bytes	Muestra el número máximo de bytes de almacenamiento de disco que se ha configurado para que lo utilicen los registros de recuperación del gestor de colas.
	Sistema de archivos de registro - bytes en uso	Muestra el número total de bytes de disco en uso para el sistema de archivos de registro.
	Sistema de archivos de registro - bytes máx	Muestra el número de bytes de disco que se han configurado para el sistema de archivos de registro.
	Registro - bytes físicos escritos	Muestra el número de bytes grabados en los registros de recuperación.

*Tabla 9. Elementos para recursos de almacenes de datos persistentes de la plataforma (continuación)*

<b>Tipo</b>	<b>Elemento</b>	<b>Descripción</b>
	Registro - bytes lógicos escritos	Muestra el número lógico de bytes grabados en los registros de recuperación.
	Registro - latencia de escritura	Muestra una medida de la latencia cuando se graba síncronamente en el registro de recuperación del gestor de colas.

*Tabla 10. Elementos para los recursos de estadísticas de uso de API*

<b>Tipo</b>	<b>Elemento</b>	<b>Descripción</b>
MQCONN y MQDISC	Recuento de MQCONN/ MQCONNX	Muestra el número de llamadas a MQCONN y MQCONNX.
	Recuento de MQCONN/ MQCONNX anómalo	Muestra el número de llamadas con error a MQCONN y MQCONNX.
	Conexiones simultáneas - marca de límite superior	Muestra el número máximo de conexiones simultáneas en el intervalo de estadísticas actual.
	Recuento de MQDISC	Muestra el número de llamadas a MQDISC.
MQOPEN y MQCLOSE	Recuento de MQOPEN	Muestra el número de llamadas a MQOPEN.
	Recuento de MQOPEN anómalo	Muestra el número de llamadas con error a MQOPEN.
	Recuento de MQCLOSE	Muestra el número de llamadas a MQCLOSE.
	Recuento de MQCLOSE anómalo	Muestra el número de llamadas con error a MQCLOSE.
MQINQ y MQSET	Recuento de MQINQ	Muestra el número de llamadas a MQINQ.
	Recuento de MQINQ anómalo	Muestra el número de llamadas con error a MQINQ.
	Recuento de MQSET	Muestra el número de llamadas a MQSET.
	Recuento de MQSET anómalo	Muestra el número de llamadas con error a MQSET.

<i>Tabla 10. Elementos para los recursos de estadísticas de uso de API (continuación)</i>		
<b>Tipo</b>	<b>Elemento</b>	<b>Descripción</b>
MQPUT	Total de intervalos de recuento de MQPUT/MQPUT1	Muestra el número de llamadas a MQPUT y MQPUT1.
	Total de intervalos de recuento de bytes de MQPUT/MQPUT1	Muestra los bytes totales de datos que las llamadas transfieren a MQPUT y MQPUT1.
	Recuento de MQPUT de mensajes no persistentes	Muestra el número de mensajes no persistentes que MQPUT transfiere.
	Recuento de MQPUT de mensajes persistentes	Muestra el número de mensajes persistentes que MQPUT transfiere.
	Recuento de MQPUT anómalo	Muestra el número de llamadas con error a MQPUT.
	Recuento de MQPUT1 de mensajes no persistentes	Muestra el número de mensajes no persistentes que MQPUT1 transfiere.
	Recuento de MQPUT1 de mensajes persistentes	Muestra el número de mensajes persistentes que MQPUT1 transfiere.
	Recuento de MQPUT1 anómalo	Muestra el número de llamadas con error a MQPUT1.
	Transferir mensaje no persistente - recuento de bytes	Muestra el número de bytes transferidos a mensajes no persistentes.
	Transferir mensaje persistente - recuento de bytes	Muestra el número de bytes transferidos a mensajes persistentes.
	Recuento de MQSTAT	Muestra el número de llamadas a MQSTAT.
	Recuento de MQSTAT con error	Muestra el número de llamadas con error a MQSTAT.
MQGET	Total de intervalos de obtención destructiva - recuento	Número de mensajes que MQGET elimina de las colas.
	Total de intervalos de obtención destructiva - recuento de bytes	Bytes de datos que MQGET elimina de las colas.
	Obtención destructiva de mensajes no persistentes - recuento	Número de mensajes no persistentes que MQGET elimina de las colas.

Tabla 10. Elementos para los recursos de estadísticas de uso de API (continuación)

Tipo	Elemento	Descripción
	Obtención destructiva de mensajes persistentes - recuento	Número de mensajes persistentes que MQGET elimina de las colas.
	MQGET anómalo - recuento	Muestra el número de llamadas con error a MQGET.
	Se han obtenido mensajes no persistentes - recuento de bytes	Muestra un recuento de bytes de mensajes no persistentes que se devuelven a MQGET.
	Se han obtenido mensajes persistentes - recuento de bytes	Muestra un recuento de bytes de mensajes persistentes que se devuelven a MQGET.
	Examen de mensajes no persistentes - recuento	Muestra un recuento de mensajes no persistentes que se han examinado.
	Examen de mensajes persistentes - recuento	Muestra un recuento de mensajes persistentes que se han examinado.
	Recuento de exámenes anómalo	Muestra un recuento de mensajes examinados que han fallado.
	Examen de mensajes no persistentes - recuento de bytes	Muestra el número de bytes de mensajes no persistentes que se han examinado.
	Examen de mensajes persistentes - recuento de bytes	Muestra el número de bytes de mensajes persistentes que se han examinado.
	Recuento de mensajes caducados	Muestra un recuento de mensajes caducados.
	Recuento de colas depuradas	Muestra un recuento de colas que se han depurado.
	Recuento de MQCB	Muestra el número de llamadas a MQCB.
	Recuento de MQCB anómalo	Muestra el número de llamadas con error a MQCB.
	Recuento de MQCTL	Muestra el número de llamadas a MQCTL.
	Recuento de MQCTL con error	Muestra el número de llamadas con error a MQCTL.
Confirmar y retrotraer	Recuento de Commit	Muestra el número de llamadas a MQCMIT.

<i>Tabla 10. Elementos para los recursos de estadísticas de uso de API (continuación)</i>		
<b>Tipo</b>	<b>Elemento</b>	<b>Descripción</b>
	Recuento de confirmaciones con error	Muestra el número de llamadas con error a MQCMIT.
	Recuento de retrotracciones	Muestra el número de llamadas a MQBACK.
Suscribir	Crear recuento de suscripciones duraderas	Muestra el número de llamadas a MQSUB para crear suscripciones duraderas.
	Alterar recuento de suscripciones duraderas	Muestra el número de llamadas a MQSUB para alterar suscripciones duraderas.
	Reanudar recuento de suscripciones duraderas	Muestra el número de llamadas a MQSUB para reanudar suscripciones duraderas.
	Crear recuento de suscripciones no duraderas	Muestra el número de llamadas a MQSUB para crear suscripciones no duraderas.
	Recuento para alterar suscripciones no duraderas	Muestra el número de llamadas a MQSUB para alterar suscripciones no duraderas.
	Recuento para reanudar suscripciones no duraderas	Muestra el número de llamadas a MQSUB para reanudar suscripciones no duraderas.
	Crear/alterar/reanudar recuento de suscripciones anómalo	Muestra el número de llamadas con error a MQSUBRQ para crear, alterar o reanudar las suscripciones.
	Suprimir recuento de suscripciones duraderas	Muestra el número de llamadas a MQSUB para suprimir suscripciones duraderas.
	Suprimir recuento de suscripciones no duraderas	Muestra el número de llamadas a MQSUB para suprimir suscripciones no duraderas.
	Recuento de anomalías de supresión de suscripción	Muestra el número de llamadas a MQSUB para suprimir suscripciones.
	Recuento de MQSUBRQ	Muestra el número de llamadas a MQSUBRQ.

<i>Tabla 10. Elementos para los recursos de estadísticas de uso de API (continuación)</i>		
<b>Tipo</b>	<b>Elemento</b>	<b>Descripción</b>
	Recuento de MQSUBRQ anómalo	Muestra el número de llamadas con error MQSUBRQ.
	Suscriptor duradero - marca de límite superior	Muestra el número máximo de suscripciones duraderas en el intervalo de estadísticas actual.
	Suscriptor duradero - marca de límite inferior	Muestra el número mínimo de suscripciones duraderas en el intervalo de estadísticas actual.
	Suscriptor no duradero - marca de límite superior	Muestra el número máximo de suscripciones no duraderas en el intervalo de estadísticas actual.
	Suscriptor no duradero - marca de límite inferior	Muestra el número mínimo de suscripciones no duraderas en el intervalo de estadísticas actual.
Publicar	Total de intervalos de MQPUT/MQPUT1 del tema	El número de mensajes transferidos a los temas.
	Total de intervalos de transferencia de bytes del tema	El número de bytes de mensajes transferidos a los temas.
	Publicado en suscriptores - recuento de mensajes	Muestra el número de mensajes que se publican a los suscriptores.
	Publicado en suscriptores - recuento de bytes	Muestra el recuento de bytes de mensajes que se publican a los suscriptores.
	No persistentes - recuento de MQPUT/MQPUT1 del tema	Muestra el número de mensajes no persistentes que se transfieren a los temas.
	Persistentes - recuento de MQPUT/MQPUT1 del tema	Muestra el número de mensajes persistentes que se transfieren a los temas.
	Recuento de MQPUT/MQPUT1 del tema anómalo	Muestra el número de intentos fallidos para transferir a un tema.

<i>Tabla 11. Elementos para los recursos de estadísticas de uso de la API por cola</i>		
<b>Tipo</b>	<b>Elemento</b>	<b>Descripción</b>
MQOPEN y MQCLOSE	Recuento de MQOPEN	Muestra el número de llamadas a MQOPEN.

Tabla 11. Elementos para los recursos de estadísticas de uso de la API por cola (continuación)

Tipo	Elemento	Descripción
	Recuento de MQCLOSE	Muestra el número de llamadas a MQCLOSE.
MQINQ y MQSET	Recuento de MQINQ	Muestra el número de llamadas a MQINQ.
	Recuento de MQSET	Muestra el número de llamadas a MQSET.
MQPUT y MQPUT1	recuento de MQPUT/MQPUT1	Muestra el número de llamadas a MQPUT y MQPUT1.
	recuento de bytes de MQPUT	Muestra los bytes totales de datos que las llamadas transfieren a MQPUT y MQPUT1.
	recuento de mensajes no persistentes de MQPUT	Muestra el número de mensajes no persistentes que MQPUT transfiere.
	recuento de mensajes persistentes de MQPUT	Muestra el número de mensajes persistentes que MQPUT transfiere.
	recuento de mensajes no persistentes de MQPUT1	Muestra el número de mensajes no persistentes que MQPUT1 transfiere.
	recuento de mensajes persistentes de MQPUT1	Muestra el número de mensajes persistentes que MQPUT1 transfiere.
	Recuento de bytes no persistentes	Muestra el número de bytes transferidos a mensajes no persistentes.
	Recuento de bytes persistentes	Muestra el número de bytes transferidos a mensajes persistentes.
	Transferencias evitadas de cola	
	Bytes evitados de cola	
	Contención de bloqueo	
MQGET	recuento de MQGET	
	recuento de bytes de MQGET	
	Recuento de mensajes no persistentes de MQGET destructivos	Número de mensajes no persistentes que MQGET elimina de la cola.
	Recuento de mensajes persistentes de MQGET destructivos	Número de mensajes persistentes que MQGET elimina de la cola.




*Tabla 11. Elementos para los recursos de estadísticas de uso de la API por cola (continuación)*

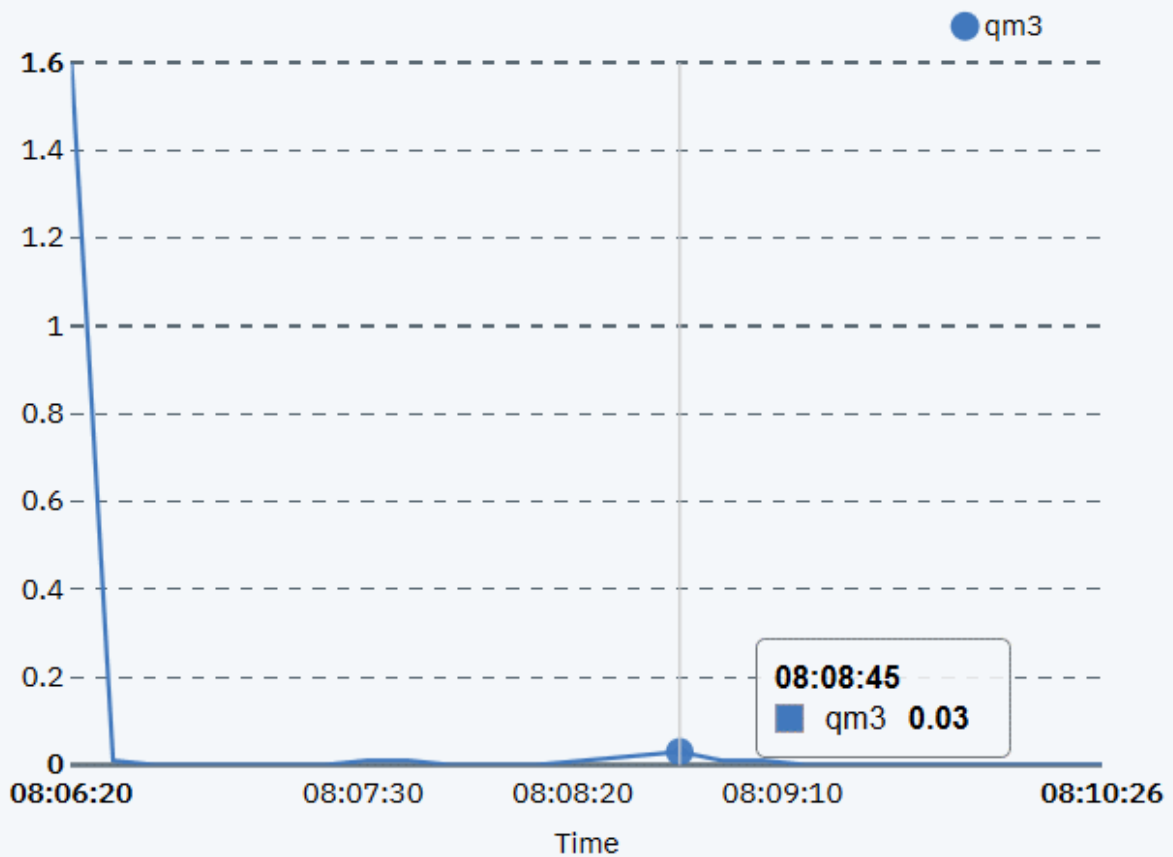
<b>Tipo</b>	<b>Elemento</b>	<b>Descripción</b>
	Recuento de bytes no persistentes de MQGET destructivos	Muestra un recuento de bytes de mensajes no persistentes que se devuelven a MQGET.
	Recuento de bytes persistentes de MQGET destructivos	Muestra un recuento de bytes de mensajes persistentes que se devuelven a MQGET.
	recuento de mensajes no persistentes de examen de MQGET	Muestra un recuento de mensajes no persistentes que se han examinado.
	recuento de mensajes persistentes de examen de MQGET	Muestra un recuento de mensajes persistentes que se han examinado.
	recuento de bytes no persistentes de examen de MQGET	Muestra el número de bytes de mensajes no persistentes que se han examinado.
	recuento de bytes persistentes de examen de MQGET	Muestra el número de bytes de mensajes persistentes que se han examinado.
	Mensajes caducados	Muestra un recuento de mensajes caducados.
	Recuento de colas depuradas	Muestra un recuento de colas que se han depurado.
	Promedio de tiempo de cola	
	Hora de la cola	

- f) Seleccione un gestor de colas para supervisar y especifique el color para visualizar información para ese gestor de colas. Pulse **Añadir** para añadir más gestores de colas. Puede especificar hasta cinco gestores de colas.
- g) Pulse **Guardar**.

## Resultados

Después de configurar el widget, transcurre un breve lapso de tiempo antes de que se visualicen los datos en el diagrama. Los datos se muestran a lo largo de un eje de tiempo. Cada punto de datos representa el final del periodo de 10 segundos sobre el que se recopilan los datos. Puede pasar el cursor por encima de los puntos de datos del gráfico para ver información detallada tal como se muestra en el ejemplo siguiente: 

## User CPU time - percentage estimate for que...




### V 9.1.0 Configuración de diseños de panel de instrumentos

Un panel de instrumentos es un contenedor de IBM MQ Console en el que se muestran los widgets. Puede crear varias pestañas de panel de instrumentos para mostrar distintas selecciones de información.

#### Acerca de esta tarea

Puede configurar cada pestaña del panel de instrumentos pulsando la flecha situada junto al nombre de

pestaña . Puede cambiar el nombre de pestaña y añadir una descripción para la pestaña. También puede configurar la cantidad de columnas que tiene la pestaña.

Puede configurar el diseño de los widgets dentro de una pestaña de panel de instrumentos arrastrando y soltando los widgets.

#### Procedimiento

- [“Creación y supresión de pestañas de panel de instrumentos” en la página 131](#)
- [“Importación y exportación de diseños de panel de instrumentos” en la página 131](#)

## **V 9.1.0 Creación y supresión de pestañas de panel de instrumentos**

Puede crear automáticamente una pestaña de panel de instrumentos que muestre información sobre un gestor de colas local específico. Puede crear y suprimir manualmente las pestañas del panel de instrumentos.

### **Acerca de esta tarea**

Cuando crea automáticamente una pestaña del panel de instrumentos para mostrar información sobre un gestor de colas local específico, se añaden automáticamente los widgets siguientes:


- Widget de colas
- Widget de canales de conexión cliente
- Widget de canales
- Widget de escuchas
- Widget de suscripciones
- Widget de temas
- Widget de información de autenticación

### **Procedimiento**

- Para crear una pestaña de panel de instrumentos:
  - a) Pulse el icono más situado junto a las pestañas del panel de control existente



- b) Escriba un nombre para la nueva pestaña.
  - c) Opcional: Escriba una descripción para la nueva pestaña.
  - d) Pulse **Añadir**.
- Para crear automáticamente una pestaña de panel de instrumentos para un gestor de colas específico:
    - a) Seleccione el gestor de colas en el widget de gestor de colas local.
    - b) Seleccione ... > **Añadir nueva pestaña del panel de instrumentos**  
Se crea una nueva pestaña del panel de instrumentos. La pestaña tiene el nombre del gestor de colas.
  - Para suprimir una pestaña del panel de instrumentos:

- a) Pulse la flecha situada junto al nombre de pestaña del panel de instrumentos  .
- b) Seleccione **Suprimir pestaña**.
- c) Confirme que desea suprimir la pestaña del panel de instrumentos pulsando **Suprimir**. Se suprime la pestaña.

## **V 9.1.0 Importación y exportación de diseños de panel de instrumentos**

Puede guardar un diseño de panel de instrumentos exportándolo desde IBM MQ Console. Puede importar un diseño de panel de instrumentos guardado en IBM MQ Console.


### **Acerca de esta tarea**

Cuando exporta un panel de instrumentos, crea un archivo .json en el disco local. Posteriormente, puede importar el archivo .json a un panel de instrumentos para volver a crear el diseño. Cuando se importa un diseño de panel de instrumentos, puede elegir añadir las pestañas importadas a un diseño de panel de


instrumentos existente. O bien, puede sustituir el diseño de panel de instrumentos existente por el diseño importado.

## Procedimiento

- Para exportar un diseño del panel de instrumentos:

- a) Pulse el icono de menú del panel de instrumentos .
- b) Seleccione **Exportar panel de instrumentos**.  
El archivo se guarda en la carpeta de descarga del navegador.

- Para importar un diseño de panel de instrumentos:

- a) Pulse el icono de menú del panel de instrumentos .
- b) Seleccione **Importar panel de instrumentos**.  
Se abre la ventana Importar configuración de panel de instrumentos.
- c) Pulse **Examinar** y busque la ubicación del archivo que contiene la configuración.
- d) Elija cómo importar las pestañas del panel de instrumentos:  
Puede elegir entre las opciones siguientes:
  - **Añadir pestañas panel de instrumentos importado al panel de instrumentos existente**
  - **Sustituir panel de instrumentos existente con las pestañas del panel de instrumentos importado**
- e) Haga clic en **Importar**.  
Se importan las pestañas del panel de instrumentos.





## Controles del panel de control

Utilice los controles en la parte superior del panel de instrumentos para habilitar el rastreo de IBM MQ Console, acceder a la ayuda en línea, ver información sobre IBM MQ Console y finalizar sesión de IBM MQ Console.

## Acerca de esta tarea

Los controles del panel de instrumentos se aplican a las características generales de IBM MQ Console.

## Procedimiento

- Acceda a IBM Documentation para IBM MQ pulsando el icono de ayuda .
- Pulse el icono de menú  para importar, exportar o restablecer el panel de instrumentos. Para obtener más información, consulte [“Configuración de diseños de panel de instrumentos”](#) en la página 130.
- Pulse el icono de valores  para habilitar o inhabilitar el rastreo de diagnóstico para IBM MQ Console.
- Consulte la información sobre IBM MQ Console, incluida la información sobre el usuario que ha iniciado la sesión, pulsando el icono de persona  y seleccionando **Acerca de**.
- Finalice sesión de IBM MQ Console pulsando **Fin de sesión**.

Esta opción no está disponible si ha iniciado sesión en IBM MQ Console mediante un certificado de cliente o si se utiliza el ejemplo de configuración de seguridad `no_security.xml`. Para obtener más información acerca de las configuraciones de seguridad, consulte [Configuración de la seguridad de IBM MQ Console](#).

## Atajos del teclado

Puede utilizar los accesos directos de teclado para trabajar con IBM MQ Console.

En la tabla siguiente se enumeran los accesos directos disponibles. Todos los accesos directos se utilizan desde dentro de un widget.

Clave	Acción
p	Ver las propiedades del objeto seleccionado
c	Crear un objeto nuevo
mayús-d	Suprimir el objeto seleccionado
a	Seleccionar todos los objetos de un widget
mayús-a	Anular selección de objetos

Windows

Linux

## Administración utilizando IBM MQ Explorer

IBM MQ Explorer le permite realizar la administración local o remota de la red desde un sistema que ejecute Windows o Linux (solo x86-64).

IBM MQ for Windows e IBM MQ para Linux x86-64 proporcionan una interfaz de administración llamada IBM MQ Explorer para realizar tareas de administración, como alternativa al uso de los mandatos de control o MQSC. [Comparación de juegos de mandatos](#) muestra lo que puede hacer utilizando IBM MQ Explorer.

IBM MQ Explorer le permite realizar la administración local o remota de la red desde un sistema que ejecute Windows, o Linux x86-64, apuntando IBM MQ Explorer a los gestores de colas y los clústeres en los que esté interesado. Se puede conectar de forma remota a los gestores de colas que se ejecutan en cualquier plataforma soportada incluyendo z/OS, lo que permite ver, explorar y modificar toda la red troncal de mensajería desde la consola.

Para configurar gestores de colas remotas de IBM MQ para que IBM MQ Explorer los pueda administrar, consulte [“Software de requisito previo y definiciones para IBM MQ Explorer”](#) en la página 135.

Le permite realizar tareas, normalmente asociadas a la configuración y el ajuste preciso del entorno de trabajo para IBM MQ, ya sea de forma local o remota dentro de un dominio de sistema Windows o Linux x86-64.

En Linux, puede que IBM MQ Explorer no se inicie correctamente si dispone de una instalación Eclipse. Si esto sucediera, inicie IBM MQ Explorer utilizando un ID de usuario diferente del que utiliza para la otra instalación de Eclipse.

En Linux, para iniciar correctamente IBM MQ Explorer, debe poder grabar un archivo en el directorio de inicio y debe existir un directorio de inicio.

IBM MQ Explorer se puede instalar ya sea como parte de la instalación del producto (consulte [Instalación y desinstalación de IBM MQ](#)) o desde la IBM MQ Explorer descarga disponible desde Fix Central (consulte [Instalación y desinstalación de IBM MQ Explorer como una aplicación autónoma en Linux y Windows](#)).

Puede utilizar IBM MQ Explorer para realizar tareas de administración utilizando una serie de vistas de contenido y diálogos de propiedades. También puede ampliar IBM MQ Explorer escribiendo uno o varios plugins de Eclipse.

### Tareas IBM MQ Explorer

Con IBM MQ Explorer, puede realizar las tareas siguientes:

- Crear y suprimir un gestor de colas (solo en la máquina local).
- Iniciar y detener un gestor de colas (solo en la máquina local).
- Definir, visualizar y modificar las definiciones de objetos de IBM MQ, como colas, temas y canales.
- Examinar los mensajes de una cola.
- Iniciar y detener un canal.
- Ver información del estado de un canal, una escucha, una cola u objetos servicio.
- Ver los gestores de colas de un clúster.
- Comprobar qué aplicaciones, usuarios o canales tienen una determinada cola abierta.
- Crear un nuevo clúster de gestor de colas utilizando el asistente para Crear un nuevo clúster.
- Añadir un gestor de colas a un clúster utilizando el asistente para Añadir un gestor de colas a un clúster.
- Gestionar el objeto de información de autenticación, que se utiliza con la seguridad de canal de TLS (seguridad de la capa de transporte).
- Crear y suprimir iniciadores de canal, supervisores desencadenantes y escuchas.
- Iniciar o detener los servidores de mandatos, los iniciadores de canal, los supervisores de activación y las escuchas.
- Configurar servicios específicos para que se inicien automáticamente cuando se inicie un gestor de colas.
- Modificar las propiedades de los gestores de colas.
- Cambiar el gestor de colas local predeterminado.
- Invocar la GUI de `strmqikm` (ikeyman) para gestionar certificados TLS, asociar certificados a gestores de colas y configurar y establecer almacenes de certificados (solo en la máquina local).
- Crear objetos de JMS a partir de objetos de IBM MQ y objetos de IBM MQ a partir de objetos de JMS.
- Crear una Fábrica de conexiones JMS para cualquiera de los tipos soportados actualmente.
- Modificar los parámetros de cualquier servicio, como el número de puerto TCP de un escucha o el nombre de una cola de iniciador de canal.
- Iniciar o detener el rastreo de servicio.

### Vistas de contenido y diálogos de propiedad

Realice tareas de administración utilizando una serie de Vistas de contenido y Diálogos de propiedades.

#### Vista de Contenido

Una vista de Contenido es un panel que puede mostrar lo siguiente:

- Atributos y opciones administrativas relativos al propio producto IBM MQ.
- Atributos y opciones administrativas relativos a uno o más objetos relacionados.
- Atributos y opciones administrativas de un clúster.

#### Diálogos de propiedades

Un diálogo de propiedades es un panel que muestra atributos relativos a un objeto en una serie de campos, algunos de los cuales se pueden editar.

Navegue por IBM MQ Explorer utilizando la vista de navegador. El navegador le permite seleccionar la vista de Contenido que necesite.

## Ampliación de IBM MQ Explorer

IBM MQ Explorer presenta la información en un estilo coherente con el de la infraestructura de Eclipse y las aplicaciones de plug-in que soporta Eclipse.

Mediante la ampliación de IBM MQ Explorer, los administradores del sistema tienen la posibilidad de personalizar IBM MQ Explorer para mejorar su forma de administrar IBM MQ.

Para obtener más información, consulte [Ampliación de MQ Explorer](#).

## Decidir si utilizar IBM MQ Explorer

A la hora de decidir si desea utilizar IBM MQ Explorer en su instalación, tenga en cuenta la información que figura en este tema.

Debe tener en cuenta los puntos siguientes:

### Nombres de objetos

Si utiliza nombres en minúsculas para gestores de colas y otros objetos con IBM MQ Explorer, cuando trabaje con los objetos utilizando mandatos MQSC, debe encerrar los nombres de objetos entre comillas simples o IBM MQ no los reconoce.

### Gestores de colas grandes

IBM MQ Explorer funciona mejor con gestores de colas pequeños. Si tiene un gran número de objetos en un solo gestor de colas, puede sufrir retrasos mientras IBM MQ Explorer extrae la información necesaria para presentarla en una vista.

### Clústeres

Los clústeres de IBM MQ pueden potencialmente contener cientos o miles de gestores de colas. IBM MQ Explorer presenta los gestores de colas de un clúster mediante una estructura de árbol. El tamaño físico de un clúster no afecta drásticamente la velocidad de IBM MQ Explorer, ya que IBM MQ Explorer no se conecta a los gestores de colas del clúster hasta que el usuario los selecciona.

## Configurar IBM MQ Explorer

En esta sección se describen los pasos que debe realizar para configurar IBM MQ Explorer.

- [“Software de requisito previo y definiciones para IBM MQ Explorer” en la página 135](#)
- [“seguridad para IBM MQ Explorer” en la página 136](#)
- [“Mostrar y ocultar gestores de colas y clústeres en IBM MQ Explorer” en la página 139](#)
- [“Pertenencia a un clúster y IBM MQ Explorer” en la página 140](#)
- [“Conversión de datos para IBM MQ Explorer” en la página 141](#)

## Software de requisito previo y definiciones para IBM MQ Explorer

Asegúrese de cumplir los siguientes requisitos antes de intentar utilizar el IBM MQ Explorer.

El IBM MQ Explorer sólo puede conectarse con gestores de colas remotos utilizando el protocolo de comunicaciones TCP/IP.

Compruebe que:

1. Se está ejecutando un servidor de mandatos en cada gestor de colas administrado de forma remota.
2. Se está ejecutando un objeto de escucha TCP/IP adecuado en cada gestor de colas remoto. Este objeto puede ser el escucha de IBM MQ o, en sistemas UNIX and Linux, el daemon inetd.
3. Existe un canal de conexión de servidor, cuyo nombre predeterminado es SYSTEM.ADMIN.SVRCONN, en todos los gestores de colas remotos.

Puede crear el canal utilizando el siguiente mandato MQSC:

```
DEFINE CHANNEL (SYSTEM.ADMIN.SVRCONN) CHLTYPE (SVRCONN)
```

Este mandato crea una definición de canal básica. Si desea una definición más compleja (por ejemplo, para configurar la seguridad), necesita parámetros adicionales. Para obtener más información, consulte [DEFINE CHANNEL](#).

4. Debe existir la cola del sistema, SYSTEM.MQEXPLORER.REPLY.MODEL.

## seguridad para IBM MQ Explorer

Si está utilizando IBM MQ en un entorno en el que es importante para usted controlar el acceso de usuarios a determinados objetos, tal vez necesite considerar los aspectos de seguridad que implica la utilización de IBM MQ Explorer.

### **Autorización para utilizar IBM MQ Explorer**

Cualquier usuario puede utilizar IBM MQ Explorer, pero se requieren ciertas autorizaciones para conectarse y acceder a los gestores de colas así como para gestionarlos.

Para realizar tareas administrativas locales utilizando IBM MQ Explorer, un usuario debe tener la autorización necesaria para realizar las tareas de administración. Si el usuario es miembro del grupo mqm, el usuario tiene autorización para realizar todas las tareas administrativas locales.

Para conectarse a un gestor de colas remoto y realizar tareas administrativas remotas utilizando IBM MQ Explorer, el usuario que ejecuta IBM MQ Explorer debe tener las siguientes autorizaciones:

- Autorización CONNECT en el objeto de gestor de colas de destino
- Autorización INQUIRE en el objeto de gestor de colas de destino
- Autorización DISPLAY para el objeto de gestor de colas de destino
- Autorización INQUIRE para la cola SYSTEM.MQEXPLORER.REPLY.MODEL
- Autorización DISPLAY para la cola SYSTEM.MQEXPLORER.REPLY.MODEL
- Autorización INPUT (get) para la cola SYSTEM.MQEXPLORER.REPLY.MODEL
- Autorización OUTPUT (put) para la cola, SYSTEM.MQEXPLORER.REPLY.MODEL
- Autorización OUTPUT (put) para la cola SYSTEM.ADMIN.COMMAND.QUEUE
- Autorización INQUIRE en la cola SYSTEM.ADMIN.COMMAND.QUEUE
- Autorización para realizar la acción seleccionada

**Nota:** La autorización INPUT está relacionada con la entrada al usuario desde una cola (una operación de obtener). Una autorización OUTPUT está relacionada con la salida desde el usuario a una cola (una operación de transferir).

Para conectarse a un gestor de colas remoto en IBM MQ for z/OS y realizar tareas administrativas remotas utilizando IBM MQ Explorer, debe proporcionarse lo siguiente:

- Un perfil RACF para la cola del sistema, SYSTEM.MQEXPLORER.REPLY.MODEL
- Un perfil RACF para las colas, AMQ.MQEXPLORER.\*

Además, el usuario que ejecute IBM MQ Explorer debe tener las siguientes autorizaciones:

- Autorización RACF UPDATE para la cola del sistema, SYSTEM.MQEXPLORER.REPLY.MODEL
- Autorización RACF UPDATE para las colas, AMQ.MQEXPLORER.\*
- Autorización CONNECT en el objeto de gestor de colas de destino
- Autorización para realizar la acción seleccionada
- Autorización READ para todos los perfiles hlq.DISPLAY.object de la clase MQCMDS

Para obtener información sobre cómo otorgar autorización sobre objetos de IBM MQ, consulte [Otorgar acceso a un objeto de IBM MQ en sistemas UNIX o Linux y Windows](#).



Si un usuario intenta realizar una operación para la que no está autorizado, el gestor de colas de destino invoca procedimientos de error de autorización y la operación no se realiza correctamente.

El filtro predeterminado en IBM MQ Explorer es mostrar todos los objetos de IBM MQ. Si hay algún objeto de IBM MQ para el que un usuario no tiene autorización DISPLAY, se generan errores de autorización. Si se están registrando sucesos de autorización, restrinja el rango de objetos que se muestran a aquellos para los que el usuario tiene autorización DISPLAY.

### ***Seguridad de la conexión a gestores de colas remotos desde IBM MQ Explorer***

Debe proteger el canal entre IBM MQ Explorer y cada gestor de colas remoto.

IBM MQ Explorer se conecta a los gestores de colas remotos como una aplicación cliente MQI. Esto significa que cada gestor de colas remoto debe tener una definición de un canal de conexión de servidor y un escucha TCP/IP adecuado. Si no protege el canal de conexión del servidor, es posible que una aplicación maliciosa preparada para ello se conecte al mismo canal de conexión de servidor y obtenga acceso a los objetos de gestor de colas con autorización ilimitada. Para proteger el canal de conexión del servidor, especifique un valor, que no sea dejarlo en blanco, para el atributo MCAUSER del canal, utilice los registros de autenticación de canal o bien utilice una salida de seguridad.

**El valor predeterminado del atributo MCAUSER es el ID de usuario local.** Si especifica un nombre de usuario que no sea un blanco como el atributo MCAUSER del canal de conexión de servidor, todos los programas que se conecten al gestor de colas utilizando este canal se ejecutan con la identidad del usuario nombrado y tiene el mismo nivel de autorización. Esto no ocurre si utiliza registros de autenticación de canal.

### ***Utilización de una salida de seguridad con IBM MQ Explorer***

Puede especificar una salida de seguridad predeterminada y salidas de seguridad específicas del gestor de colas utilizando IBM MQ Explorer.

Puede definir una salida de seguridad predeterminada, que se puede utilizar para todas las conexiones de cliente nuevas desde IBM MQ Explorer. Esta salida predeterminada se puede alterar temporalmente en el momento en que se realiza una conexión. También puede definir una salida de seguridad para un gestor de colas individual o para un conjunto de gestores de colas, lo que se lleva a cabo cuando se realiza la conexión. Las salidas se especifican utilizando IBM MQ Explorer. Para obtener más información, consulte la ayuda de IBM MQ Explorer.

### ***Utilización de IBM MQ Explorer para conectarse a un gestor de colas remoto utilizando canales MQI habilitados para TLS***

IBM MQ Explorer se conecta a los gestores de colas remotos utilizando un canal MQI. Si desea proteger el canal MQI con el protocolo de seguridad TLS, debe establecer el canal utilizando una tabla de definiciones de canal de cliente.

Para obtener información sobre cómo establecer un canal MQI utilizando una tabla de definición de canal de cliente, consulte [Visión general de IBM MQ MQI clients](#).

Cuando haya establecido el canal utilizando una tabla de definiciones de canal de cliente, puede utilizar el IBM MQ Explorer para conectarse a un gestor de colas remoto utilizando canales MQI habilitados para TLS, tal como se describe en [“Tareas en el sistema en el que se aloja el gestor de colas remoto”](#) en la página 137 y en [“Tareas en el sistema que aloja IBM MQ Explorer”](#) en la página 138.

## **Tareas en el sistema en el que se aloja el gestor de colas remoto**

En el sistema en el que se aloje el gestor de colas remoto, efectúe las tareas siguientes:

1. Defina un par de canales de conexión de servidor y conexión de cliente y especifique el valor adecuado para el atributo `SSLCIPH` en la conexión de servidor en ambos canales. Para obtener más información sobre el atributo `SSLCIPH`, consulte [Protección de canales con TLS](#)
2. Envíe la tabla de definición de canal `AMQCLCHL.TAB`, que se encuentra en el directorio `@ipcc` del gestor de colas, al sistema que aloja IBM MQ Explorer.
3. Inicie un escucha TCP/IP en un puerto designado.

4. Coloque los certificados de CA y TLS personales en el directorio SSL del gestor de colas:

- /var/mqm/qmgrs/+QMNAME+/SSL para sistemas UNIX and Linux
  - C:\Archivos de programa\IBM\MQ\qmgrs\+QMNAME+\SSL para sistemas Windows
- Donde +QMNAME+ es una señal que representa el nombre del gestor de colas.

5. Cree un archivo de base de datos de claves de tipo CMS denominado key . kdb. Oculte la contraseña en un archivo, seleccionando la opción en la GUI de **strmqikm** (iKeyman) o utilizando la opción -stash con los mandatos **runmqckm** .

6. Añada los certificados CA a la base de datos de claves creada en el paso anterior.

7. Importe el certificado personal del gestor de colas a la base de datos de claves.

Para obtener información más detallada sobre cómo trabajar con TLS en sistemas Windows, consulte [Trabajar con TLS en UNIX, Linux, and Windows](#).

## Tareas en el sistema que aloja IBM MQ Explorer

En el sistema que aloja IBM MQ Explorer, realice las tareas siguientes:

1. Cree un archivo de base de datos de claves de tipo JKS denominado key . jks. Establezca una contraseña para este archivo.

El IBM MQ Explorer utiliza los archivos de almacén de claves Java (JKS) para la seguridad TLS, por lo que el archivo de almacén de claves que se crea para configurar TLS para el IBM MQ Explorer debe coincidir con éste.

2. Añada los certificados CA a la base de datos de claves creada en el paso anterior.

3. Importe el certificado personal del gestor de colas a la base de datos de claves.

4. En sistemas Windows y Linux, inicie IBM MQ Explorer utilizando el menú del sistema, el archivo ejecutable MQExplorer o el mandato **strmqcfig**.

5. En la barra de herramientas de IBM MQ Explorer, pulse **Ventana -> Preferencias**, luego expanda **IBM MQ Explorer** y pulse **Almacenes de certificados SSL de cliente**. Entre el nombre y la contraseña del archivo JKS creado en el paso 1 de “Tareas en el sistema que aloja IBM MQ Explorer” en la página 138, tanto en el Almacén de certificados fiable como en el Almacén de certificados personal, y pulse **Aceptar**.

6. Cierre la ventana **Preferencias** y pulse el botón derecho del ratón en **Gestores de colas**. Pulse **Mostrar/ocultar gestores de colas** y luego pulse **Añadir** en la pantalla **Mostrar/ocultar gestores de colas**.

7. Escriba el nombre del gestor de colas y seleccione la opción **Conectar directamente**. Pulse Siguiente.

8. Seleccione **Utilizar tabla de definiciones de canal de cliente (CCDT)** y especifique la ubicación del archivo de tabla de canales que ha transferido del gestor de colas remoto en el paso 2 en “Tareas en el sistema en el que se aloja el gestor de colas remoto” en la página 137 en el sistema en el que se aloja el gestor de colas remoto.

9. Pulse **Finalizar**. Ahora puede acceder al gestor de colas remoto desde el IBM MQ Explorer.

## Conexión a través de otro gestor de colas con IBM MQ Explorer

IBM MQ Explorer le permite conectarse a un gestor de colas a través de un gestor de colas intermedio, al que IBM MQ Explorer ya está conectado.

En este caso, IBM MQ Explorer transfiere mensajes de mandato PCF al gestor de colas intermedio, especificando lo siguiente:

- El parámetro *NombGstColasObj* del descriptor de objeto (MQOD) como nombre del gestor de colas de destino. Para obtener más información sobre la resolución de nombres de cola, consulte [Resolución de nombres](#).
- El parámetro *IdentificadorUsuario* del descriptor de mensaje (MQMD) como el ID de usuario local.

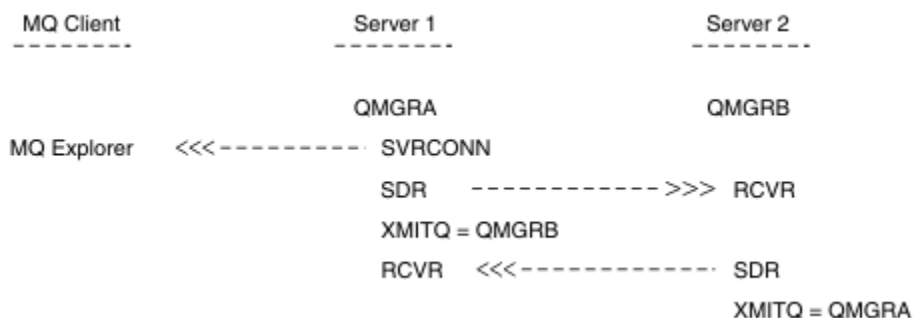
Si la conexión se utiliza después para conectar con el gestor de colas de destino mediante un gestor de colas intermediario, el ID de usuario se incorpora al parámetro *IdentificadorUsuario* del descriptor de mensaje (MQMD). Para que el escucha MCA del gestor de colas de destino acepte este mensaje, debe establecerse el atributo MCAUSER o el ID de usuario ya debe existir con autorización para transferencias.

El servidor de mandatos del gestor de colas de destino transfiere mensajes a la cola de transmisión especificando el ID de usuario del parámetro *IdentificadorUsuario* del descriptor de mensaje (MQMD). Para que esta transferencia se realice correctamente, el ID de usuario ya debe existir en el gestor de colas de destino con autorización para transferencias.

El siguiente ejemplo le muestra cómo conectar un gestor de colas, a través de un gestor de colas intermedio, a IBM MQ Explorer.

Establezca una conexión de administración remota con un gestor de colas. Verifique que:

- El gestor de colas en el servidor está activo y tiene un canal de conexión de servidor (SVRCONN) definido.
- El escucha está activo.
- El servidor de mandatos está activo.
- La cola SYSTEM.MQ EXPLORER.REPLY.MODEL se ha creado y que tiene suficiente autorización.
- Los escuchas, los servidores de mandatos y los canales emisores de los gestores de colas están iniciados.



En este ejemplo:

- IBM MQ Explorer se conecta al gestor de colas QMGRA (que se ejecuta en el Servidor1) utilizando una conexión de cliente.
- El gestor de colas QMGRB en el Servidor2 ahora puede conectarse a IBM MQ Explorer a través de un gestor de colas intermedio (QMGRA)
- Al conectarse a QMGRB con IBM MQ Explorer, seleccione QMGRA como el gestor de colas intermedio

En esta situación, no hay ninguna conexión directa a QMGRB desde IBM MQ Explorer; la conexión a QMGRB se realiza a través de QMGRA.

El gestor de colas QMGRB en el Servidor2 se conecta a QMGRA en el Servidor1 utilizando canales emisor-receptor. El canal entre QMGRA y QMGRB debe configurarse de tal manera que sea posible la administración remota; consulte [“Configuración de gestores de colas para la administración remota”](#) en la página 208.

## Mostrar y ocultar gestores de colas y clústeres en IBM MQ Explorer

IBM MQ Explorer puede mostrar más de un gestor de colas a la vez. En el panel Mostrar/ocultar gestor de colas (que se puede seleccionar desde el menú del nodo de árbol Gestores de colas), puede decidir si se visualiza información sobre otra máquina (remota). Los gestores de colas locales se detectan automáticamente.

Para mostrar un gestor de colas remoto:

1. Pulse el botón derecho (del ratón) en el nodo de árbol **Gestores de colas** y, a continuación, seleccione **Mostrar/ocultar gestores de colas**.
2. Pulse **Añadir**. Aparece el panel Mostrar/ocultar gestores de colas.
3. Escriba el nombre del gestor de colas remoto y el nombre de host o dirección IP en los campos correspondientes.

El nombre de host o dirección IP se utiliza para establecer una conexión de cliente con el gestor de colas remoto utilizando el canal de conexión de servidor predeterminado, SYSTEM.ADMIN.SVRCONN, o un canal de conexión de servidor definido por el usuario.

4. Pulse **Finalizar**.

El panel Mostrar/ocultar gestores de colas también muestra una lista de todos los gestores de colas visibles. Puede utilizar este panel para ocultar gestores de colas de la vista de navegación.

Si IBM MQ Explorer muestra un gestor de colas que es miembro de un clúster, el clúster se detecta y se muestra automáticamente.

Para exportar la lista de gestores de colas remotos desde este panel:

1. Cierre el panel Mostrar/ocultar gestores de colas.
2. Pulse el botón derecho del ratón en el nodo de árbol de **IBM MQ** superior en el panel Navegación de IBM MQ Explorer, y luego seleccione **Exportar valores de IBM MQ Explorer**
3. Pulse **IBM MQ Explorer > Valores de IBM MQ Explorer**
4. Seleccione **Información de conexión > Gestores de colas remotos**.
5. Seleccione un archivo para almacenar los valores de configuración exportados.
6. Por último, pulse **Finalizar** para exportar la información de conexión de gestor de colas remoto al archivo especificado.

Para importar una lista de gestores de colas remotos:

1. Pulse el botón derecho del ratón en el nodo de árbol de **IBM MQ** superior en el panel Navegación de IBM MQ Explorer y luego seleccione **Importar valores de IBM MQ Explorer**
2. Pulse **IBM MQ Explorer > Valores de IBM MQ Explorer**
3. Pulse **Examinar** y desplácese a la vía de acceso del archivo que contiene la información de conexión de gestor de colas remoto.
4. Pulse **Abrir**. Si el archivo contiene una lista de gestores de colas remotos, se selecciona el recuadro **Información de conexión > Gestores de colas remotos**.
5. Por último, pulse **Finalizar** para importar la información de conexión de gestor de colas remoto a IBM MQ Explorer.

## Pertenencia a un clúster y IBM MQ Explorer

IBM MQ Explorer necesita información sobre los gestores de colas que son miembros de un clúster.

Si un gestor de colas es miembro de un clúster, el nodo de árbol del clúster se llenará automáticamente.

Si los gestores de colas se convierten en miembros de clústeres mientras IBM MQ Explorer está ejecutándose, debe mantener IBM MQ Explorer con datos de administración actualizados sobre los clústeres para que éste pueda comunicarse eficazmente con ellos y mostrar información correcta sobre los clústeres cuando se le solicite. Para ello, IBM MQ Explorer necesita la siguiente información:

- El nombre de un gestor de colas de depósito
- El nombre de conexión del gestor de colas de repositorio si éste se encuentra en un gestor de colas remoto.

Con esta información, IBM MQ Explorer puede:

- Utilizar el gestor de colas de repositorio para obtener una lista de los gestores de colas del clúster.

- Administrar los gestores de colas que sean miembros del clúster y estén en plataformas y niveles de mandatos soportados.

La administración no es posible si:

- El depósito elegido deja de estar disponible. IBM MQ Explorer no cambia automáticamente a un repositorio alternativo.
- No se puede contactar con el depósito elegido a través de TCP/IP.
- El repositorio elegido está ejecutándose en un gestor de colas que se está ejecutando en una plataforma y un nivel de mandatos no soportados por IBM MQ Explorer.

Los miembros del clúster que pueden administrarse pueden ser locales o remotos (si es posible comunicar con ellos a través de TCP/IP). IBM MQ Explorer se conecta directamente con los gestores de colas locales que son miembros de un clúster, sin utilizar una conexión de cliente.

## Conversión de datos para IBM MQ Explorer

IBM MQ Explorer funciona con el CCSID 1208 (UTF-8). Esto permite a IBM MQ Explorer mostrar los datos de gestores de colas remotos correctamente. Tanto si se conecta a un gestor de colas directamente como si lo hace a través de un gestor de colas intermedio, IBM MQ Explorer requiere que todos los mensajes entrantes se conviertan a CCSID 1208 (UTF-8).

Se emite un mensaje de error si se intenta establecer una conexión entre IBM MQ Explorer y un gestor de colas con un CCSID que IBM MQ Explorer no reconoce.

Las conversaciones soportadas se describen en [Conversión de página de códigos](#).

## Windows Utilización de la aplicación Barra de tareas de IBM MQ (sólo Windows)

La aplicación Barra de tareas de IBM MQ muestra un icono en la bandeja del sistema de Windows del servidor. El icono le proporciona el estado actual de IBM MQ y un menú desde el que puede ejecutar algunas acciones simples.

En Windows, el icono de IBM MQ está en la bandeja del sistema del servidor y tiene superpuesto un símbolo de estado diferenciado con colores que puede tener uno de los siguientes significados:

### Verde

Funciona correctamente, no hay alertas en este momento

### Azul

Indeterminado; IBM MQ se está iniciando o cerrando

### Amarillo

Alerta; uno o más servicios están fallando o ya han fallado.

Para visualizar el menú, pulse el botón derecho del ratón en el icono de IBM MQ. Desde el menú puede realizar las acciones siguientes:

- Pulse **Abrir** para abrir el supervisor de alertas de IBM MQ.
- Pulse **Salir** para salir de la aplicación de barra de tareas de IBM MQ.
- Pulse **IBM MQ Explorer** para iniciar IBM MQ Explorer.
- Pulse **Detener IBM MQ** para detener IBM MQ.
- Pulse **Acerca de IBM MQ** para visualizar información sobre el supervisor de alertas de IBM MQ.

## Windows La aplicación de supervisor de alertas de IBM MQ (sólo Windows)

El supervisor de alertas de IBM MQ es una herramienta de detección de errores que identifica y registra problemas con IBM MQ en una máquina local.

El supervisor de alertas muestra información sobre el estado actual de la instalación local de un servidor de IBM MQ. También supervisa la Interfaz avanzada de configuración y energía (ACPI) de Windows y asegura que se utilicen los valores de la ACPI.

Desde el supervisor de alertas de IBM MQ, puede:

- Acceder a IBM MQ Explorer directamente
- Ver información relacionada con todas las alertas pendientes
- Concluir el servicio IBM MQ en la máquina local
- Direccionar mensajes de alerta a través de la red a una cuenta de usuario configurable o a una estación de trabajo o un servidor de Windows

## Trabajar con objetos de IBM MQ locales

---

Se pueden administrar objetos locales de IBM MQ para soportar programas de aplicación que usen la interfaz de cola de mensajes (Message Queue Interface, MQI).

### Acerca de esta tarea

En este contexto, administración local significa crear, visualizar, cambiar, copiar y suprimir objetos de IBM MQ.

Además de los enfoques descritos en esta sección, puede utilizar la IBM MQ Explorer local para administrar objetos IBM MQ. Para obtener más información, consulte [“Administración utilizando IBM MQ Explorer”](#) en la página 133.

### Procedimiento

- Utilice la información de los temas siguientes como ayuda en la administración local objetos de IBM MQ.
  - [Programas de aplicación que utilizan la MQI](#)
  - [“Administración utilizando mandatos MQSC”](#) en la página 12
  - [“Visualización y modificación de atributos del gestor de colas”](#) en la página 150
  - [“Trabajar con colas locales”](#) en la página 153
  - [“Trabajar con colas alias”](#) en la página 165
  - [“Trabajar con colas modelo”](#) en la página 167
  - [“Trabajar con servicios”](#) en la página 194
  - [“Gestión de objetos para desencadenamiento”](#) en la página 202

## Trabajar con gestores de colas

Puede utilizar los mandatos de control para iniciar y detener un gestor de colas. Puede utilizar los mandatos MQSC para visualizar o modificar los atributos del gestor de colas.

### Tareas relacionadas

[Creación de gestores de colas en Multiplatforms](#)

### Inicio de un gestor de colas

Cuando se crea un gestor de colas, debe iniciarse para habilitarlo para que pueda procesar mandatos o llamadas MQI.

### Acerca de esta tarea

Puede iniciar un gestor de colas mediante el mandato **strmqm**. Para obtener una descripción del mandato **strmqm** y sus opciones, consulte [strmqm](#).

**Windows** **Linux** O bien, en los sistemas Windows y Linux (plataformas x86 y x86-64), puede iniciar un gestor de colas utilizando IBM MQ Explorer.

**Windows** En Windows puede iniciar un gestor de colas automáticamente cuando se inicia el sistema utilizando IBM MQ Explorer. Para obtener más información, consulte [“Administración utilizando IBM MQ Explorer”](#) en la página 133.

## Procedimiento

- Para iniciar un gestor de colas mediante el mandato **strmqm**, especifique el mandato seguido del nombre del gestor de colas que desee iniciar.

Por ejemplo, para iniciar un gestor de colas llamado QMB, especifique el siguiente mandato:

```
strmqm QMB
```

**Nota:** Debe utilizar el mandato **strmqm** desde la instalación asociada al gestor de colas con el que está trabajando. Puede averiguar con qué instalación está asociado un gestor de colas utilizando el mandato `dspmqr -o installation`.

El mandato **strmqm** no devolverá el control hasta que el gestor de colas que se haya iniciado y esté preparado para aceptar solicitudes de conexión.

- **Windows** **Linux**  
Para iniciar un gestor de colas utilizando IBM MQ Explorer, realice los pasos siguientes:
  - a) Abra IBM MQ Explorer.
  - b) En la vista de Navegador, seleccione el gestor de colas.
  - c) Pulse **Iniciar**.

## Resultados

El gestor de colas se inicia.

Si el inicio del gestor de colas tarda más de unos segundos, IBM MQ emite mensajes informativos de forma intermitente que detallan el progreso del inicio.

## **Multi** Detención de un gestor de colas

Puede utilizar el mandato **endmqm** para detener un gestor de colas. Este mandato proporciona cuatro formas para detener un gestor de colas: un cierre controlado, o desactivado temporalmente, una conclusión inmediata, una conclusión preferente o una conclusión en espera. O bien, en los sistemas Windows y Linux, se puede parar un gestor de colas utilizando IBM MQ Explorer.

## Acerca de esta tarea

Existen cuatro formas para detener un gestor de colas de una sola instancia con el mandato **endmqm**:

### Cierre controlado (por desactivación temporal)

De forma predeterminada, el mandato **endmqm** realiza una conclusión por desactivación temporal del gestor de colas especificado. Un cierre por desactivación temporal espera hasta que todas las aplicaciones conectadas se hayan desconectado, por tanto podría tardar un poco en completarse.

### Conclusión inmediata

Para una conclusión inmediata, se permite que terminen las llamadas MQI que haya en ese momento, pero las llamadas nuevas fallarán. Este tipo de conclusión no espera a que las aplicaciones se desconecten del gestor de colas.

## Conclusión preferente

El gestor de colas se para de forma inmediata. Utilice este tipo de cierre solo en circunstancias excepcionales, por ejemplo, cuando un gestor de colas no se pare como resultado de un mandato **endmqm** normal.

## Conclusión de espera

Este tipo de conclusión es parecido a la conclusión controlada excepto en que el usuario sólo recupera el control después de que el gestor de colas se haya detenido.

El mandato **endmqm** detiene todas las instancias de un gestor de colas multiinstancia de la misma manera que detiene un gestor de colas de una sola instancia. Puede emitir el mandato **endmqm** en una instancia activa o en una de las instancias en espera de un gestor de colas multiinstancia. Sin embargo, debe emitir **endmqm** en la instancia activa para finalizar el gestor de colas.

**V 9.1.4** A partir de IBM MQ 9.1.4, tiene la opción de finalizar el gestor de colas en un objetivo de tiempo de un número de segundos que especifique, ya sea con o sin interrumpir las tareas básicas de mantenimiento del gestor de colas. Hay dos formas de especificar una hora de destino. La opción **-t** permite que se completen las tareas esenciales de mantenimiento del gestor de colas, lo que puede prolongar la fase de finalización del gestor de colas. La opción **-tp** interrumpe las tareas de mantenimiento del gestor de colas esenciales si es necesario para cumplir con el tiempo de destino especificado.

**V 9.1.4** Cuando se especifica un objetivo de tiempo, el tipo de conclusión de **-w**, **-i** o **-p** indica el tipo de conclusión de inicio.

**Nota:** Una conclusión **immediate** sigue siendo ordenada, que difiere de una conclusión **controlled** básicamente en la forma en la que se desactivan temporalmente las aplicaciones en ejecución. Una conclusión **immediate** sigue realizando acciones de mantenimiento como, por ejemplo compactar la colas y persistir mensajes NPMCLASS(HIGH) que podrían tardar mucho tiempo, mientras que una conclusión con tiempo limitado abandona estas acciones adicionales cuando interfieren en el cumplimiento del objetivo de tiempo.

**V 9.1.4** El gestor de colas escala el tipo de conclusión según sea necesario, en un intento por cumplir el objetivo de tiempo. Por ejemplo:

- Un objetivo de tiempo de 10 segundos **-t** que empieza en **-w** podría estar siete segundos con la inmovilización, dos segundos con la conclusión inmediata del gestor de colas, incluyendo el mantenimiento, y después realizar una conclusión inmediata sin realizar ningún otro mantenimiento:

```
endmqm -w -t 10 queue_manager
```

- Un objetivo de tiempo **-tp** de 10 segundos podría estar siete segundos con la inmovilización, dos segundos con la conclusión inmediata del gestor de colas, incluido el mantenimiento, un segundo con la conclusión inmediata sin realizar ningún otro mantenimiento y, a continuación, empezar a finalizar los procesos de IBM MQ:

```
endmqm -c -tp 10 queue_manager
```

- Un objetivo de tiempo de dos segundos **-tp** en **-i** podría estar un segundo con la conclusión inmediata del gestor de colas, incluido el mantenimiento, un segundo con la conclusión inmediata sin realizar ningún otro mantenimiento y, a continuación, empezar a finalizar los procesos de IBM MQ:

```
endmqm -i -tp 2 queue_manager
```

- Un objetivo de tiempo de un segundo en **-w** podría estar 0,1 segundos en wait (espera), por ejemplo, solo el tiempo suficiente para enviar los códigos de retorno de IBM MQ a las aplicaciones conectadas, 0,9 segundos con la conclusión inmediata del gestor de colas, incluido el mantenimiento, y después con la conclusión inmediata sin realizar ningún otro mantenimiento; a continuación, empezar a finalizar los procesos de IBM MQ:



### Atención:



- Los mensajes permanentes persistirán independientemente del tipo de conclusión utilizado (incluyendo la finalización manual de procesos IBM MQ), mientras que no se puede garantizar que los mensajes no permanentes sobrevivan cualquier tipo de conclusión.

NPMCLASS(HIGH) guarda los mensajes no permanentes de la mejor forma posible. El uso de **endmqm -t**, **endmqm -tp**, **endmqm -po** la finalización manual de procesos IBM MQ reduce las posibilidades de que los mensajes NPMCLASS (HIGH) sobrevivan a un ciclo de conclusión o reinicio de IBM MQ en comparación con **endmqm -w** o **endmqm -i**

- El tiempo combinado para finalizar y, también, reiniciar el gestor de colas puede ser más largo como resultado del uso de una conclusión más abrupta, en particular, cuando se utilizan las opciones **-p** y **-tp**.



Si el gestor de colas tiene que volver a recurrir a la finalización de procesos IBM MQ para finalizar el gestor de colas, es probable que sea necesario más reconciliación del estado del gestor de colas, cuando se reinicia el gestor de colas.

Para obtener una descripción detallada del mandato **endmqm** y sus opciones, consulte [endmqm](#).

**Consejo:** A menudo, la causa de los problemas que pueden surgir al cerrar un gestor de colas está en las aplicaciones. Por ejemplo, cuando las aplicaciones:

- No comprueban adecuadamente los códigos de retorno MQI
- No solicitan notificación de una conclusión progresiva
- Terminan sin desconectarse del gestor de colas (emitiendo una llamada MQDISC)

Si se produce un problema al intentar parar el gestor de colas, se puede salir del mandato **endmqm** con Control+C. Después, se puede ejecutar otro mandato **endmqm**, pero esta vez con un distintivo que especifique el tipo de cierre que se desea realizar.

  Como alternativa al mandato **endmqm**, en Windows y Linux, se puede parar un gestor de colas utilizando IBM MQ Explorer para llevar a cabo un cierre controlado o inmediato.

## Procedimiento

- Para detener el gestor de colas mediante el mandato **endmqm**, especifíquelo seguido del parámetro adecuado, si es necesario, y del nombre del gestor de colas que desee detener.

**Nota:** Debe utilizar el mandato **endmqm** desde la instalación asociada al gestor de colas con el que está trabajando. Para averiguar con qué instalación está asociado un gestor de colas, utilice el mandato **dspmqr** :

```
dspmqr -o installation
```

- Para realizar un cierre controlado (por desactivación temporal), ejecute el mandato **endmqm** tal y como se muestra en el ejemplo siguiente, que para un gestor de colas llamado QMB:

```
endmqm QMB
```

De forma alternativa, la ejecución del mandato **endmqm** con el parámetro **-c**, tal y como se muestra en el ejemplo siguiente, es equivalente al mandato **endmqm QMB**.

```
endmqm -c QMB
```

En ambos casos, el control se devuelve inmediatamente y no se notifica cuándo se para el gestor de colas. Si desea que el mandato espere a que todas las aplicaciones se hayan parado y el gestor de colas haya finalizado antes de devolverle el control, utilice el parámetro **-w** en su lugar, como se muestra en el ejemplo siguiente.

```
endmqm -w QMB
```


- Para llevar a cabo un cierre inmediato, escriba el mandato **endmqm** con el parámetro **-i**, como se muestra en el ejemplo siguiente:

```
endmqm -i QMB
```

- Para llevar a cabo un cierre preferente, escriba el mandato **endmqm** con el parámetro **-p**, como se muestra en el ejemplo siguiente:

```
endmqm -p QMB
```



**Atención:** Un cierre preferente puede tener consecuencias imprevisibles para las aplicaciones conectadas. No utilice esta opción a menos que todos los demás intentos de parar el gestor de colas con un mandato **endmqm** normal hayan fracasado.  Si el cierre preferente no funciona, intente [“Detención manual de un gestor de colas”](#) en la [página 146](#) en su lugar.

- Para solicitar la reconexión de cliente automática, especifique el mandato **endmqm** con el parámetro **-r**. Este parámetro tiene el efecto de restablecer la conectividad de los clientes con otros gestores de colas en el grupo de gestores de colas.

**Nota:** La finalización de un gestor de colas utilizando el mandato **endmqm** predeterminado no desencadena la reconexión automática de cliente.

- Para transferir a una instancia en espera de un gestor de colas multiinstancia después de concluir la instancia activa, especifique el mandato **endmqm** con el parámetro **-s** en la instancia activa del gestor de colas multiinstancia.
- Para finalizar la instancia en espera de un gestor de colas multiinstancia y dejar la instancia activa en ejecución, especifique el mandato **endmqm** con el parámetro **-x** en la instancia en espera del gestor de colas multiinstancia.

En Windows y Linux, para detener el gestor de colas con IBM MQ Explorer, siga los pasos siguientes:

- a) Abra IBM MQ Explorer.
- b) Seleccione el gestor de colas en la vista de navegador.
- c) Pulse **Detener**.  
Aparece el panel **Terminar gestor de colas**.
- d) Seleccione **Controlado** o **Inmediato**.
- e) Pulse **Aceptar**.  
El gestor de colas se detiene.

### Tareas relacionadas

[Aplicación de actualizaciones de nivel de mantenimiento a gestores de colas multiinstancia en AIX](#)

[Aplicación de actualizaciones de nivel de mantenimiento a gestores de colas multiinstancia en Linux](#)

[Aplicación de actualizaciones de nivel de mantenimiento a gestores de colas multiinstancia en Solaris](#)

[Aplicación de actualizaciones de nivel de mantenimiento a gestores de colas multiinstancia en Windows](#)

## Detención manual de un gestor de colas

Si fallan los métodos empleados habitualmente para detener y eliminar gestores de colas, puede recurrir a los métodos aquí descritos.

### Acerca de esta tarea

La forma estándar de detener los gestores de colas es utilizar el mandato **endmqm**, tal como se describe en [“Detención de un gestor de colas”](#) en la [página 143](#). Si no puede parar un gestor de colas de la forma

habitual, puede intentar pararlo manualmente. El modo en que lo haga dependerá de la plataforma que esté utilizando.

## Procedimiento

- **Windows**  
Para detener un gestor de colas en Windows, consulte [“Parada manual de un gestor de colas en Windows”](#) en la página 147.
- **Linux** **UNIX**  
Para detener un gestor de colas en UNIX o Linux, consulte [“Parada manual de un gestor de colas en UNIX y Linux”](#) en la página 148.

### Tareas relacionadas

[Creación y gestión de gestores de colas en Multiplatforms](#)

### Referencia relacionada

[endmqm](#)

### **Windows** Parada manual de un gestor de colas en Windows

Si no se puede parar un gestor de colas en Windows con el mandato **endmqm**, se puede intentar parar el gestor de colas manualmente terminando los procesos que están ejecutando y parando el servicio de IBM MQ.

## Acerca de esta tarea

**Consejo:** El Administrador de tareas de Windows y el mandato **tasklist** proporcionan información limitada sobre las tareas. Para obtener más información que le ayude a determinar qué procesos están relacionados con un gestor de colas concreto, puede utilizar una herramienta como *Process Explorer* (`procexp.exe`), que puede descargarse en el sitio web de Microsoft en <https://www.microsoft.com>.

Para parar un gestor de colas en Windows, siga estos pasos:

## Procedimiento

1. Enumere los nombres (ID) de los procesos que están ejecutándose actualmente utilizando el Administrador de tareas de Windows.
2. Finalice los procesos utilizando el Administrador de tareas de Windows o el mandato **taskkill**, en el orden siguiente (si están en ejecución):

*Tabla 13. Procesos Windows que hay que parar si están ejecutando*

Nombre de proceso	Descripción
AMQZMUC0	Gestor de procesos críticos
AMQZXMA0	Controlador de ejecución
AMQZFUMA	Proceso del OAM
AMQZLAA0	Agentes LQM
AMQZLSA0	Agentes LQM
AMQZMUFO	Gestor de programas de utilidad
AMQZMGRO	Controlador de procesos
AMQZMUR0	Gestor de procesos reiniciable
AMQFQPUB	Proceso de Publicación/suscripción
AMQFCXBA	Proceso de operador de intermediario

Tabla 13. Procesos Windows que hay que parar si están ejecutando (continuación)	
Nombre de proceso	Descripción
AMQRMPPA	Proceso de agrupación de procesos
AMQCRSTA	Proceso de trabajo de respuesta sin hebra
AMQCRS6B	Conexión de cliente y canal receptor LU62
AMQRRMFA	Proceso de depósito (para clústeres)
AMQPCSEA	El servidor de mandatos
RUNMQTRM	Invocar un supervisor de activación para un servidor
RUNMQDLQ	Invocar manejador de la cola de mensajes no entregados
RUNMQCHI	El proceso de iniciado de canal
RUNMQLSR	El proceso de escucha de canal
AMQXSSVN	Servidores de memoria compartida

3. Detenga el servicio IBM MQ desde **Herramientas de administración > Servicios** en el Panel de control de Windows.
4. Si ha probado todos los métodos y el gestor de colas no se ha detenido, reinicie el sistema.

### Linux → UNIX **Parada manual de un gestor de colas en UNIX y Linux**

Si no se puede parar un gestor de colas en UNIX y Linux con el mandato **endmqm**, se puede intentar parar el gestor de colas manualmente terminando los procesos en ejecución y parando el servicio IBM MQ.

#### Acerca de esta tarea

Para detener un gestor de colas en UNIX y Linux, siga los pasos siguientes.

Si detiene el gestor de colas manualmente, es posible que se tome FFST y que los archivos FDC se coloquen en `/var/mqm/errors`. Esto no debe considerarse un defecto del gestor de colas.

El gestor de colas se reiniciará normalmente, incluso después de haberlo parado con el método manual.

#### Procedimiento

1. Busque los ID de proceso de los programas del gestor de colas que estén ejecutándose aún, mediante el mandato **ps**.

Por ejemplo, si el nombre del gestor de colas es QMNAME, puede utilizar el siguiente mandato:

```
ps -ef | grep QMNAME
```

2. Finalice los procesos de gestor de colas que todavía estén en ejecución utilizando el mandato **kill**, especificando los PID descubiertos utilizando el mandato **ps**.

Para finalizar un proceso, utilice **kill -KILL <pid>** o el mandato **kill -9 <pid>** equivalente.

Tiene que trabajar a través de los PID que desea matar, uno por uno, emitiendo ese mandato cada vez.

**Importante:** Si utiliza cualquier señal que no sea **9 (SIGKILL)**, el proceso probablemente no se detendrá y obtendrá resultados imprevisibles.

Finalice los procesos en el orden siguiente:

<i>Tabla 14. Procesos de UNIX y Linux que hay que parar si están ejecutando</i>	
<b>Nombre de proceso</b>	<b>Descripción</b>
amqzmuc0	Gestor de procesos críticos
amqzma0	Controlador de ejecución
amqzfuma	Proceso del OAM
amqzlaa0	Agentes LQM
amqzlsa0	Agentes LQM
amqzmuf0	Gestor de programas de utilidad
amqzmur0	Gestor de procesos reiniciable
amqzmgr0	Controlador de procesos
amqfqpub	Proceso de Publicación/suscripción
amqfcxba	Proceso de operador de intermediario
amqrmppa	Proceso de agrupación de procesos
amqcrsta	Proceso de trabajo de respuesta sin hebra
amqcrs6b	Conexión de cliente y canal receptor LU62
amqrrmfa	Proceso de depósito (para clústeres)
amqpcsea	El servidor de mandatos
runmqtrm	Invocar un supervisor de activación para un servidor
runmqdlq	Invocar manejador de la cola de mensajes no entregados
runmqchi	El proceso de iniciado de canal
runmqlsr	El proceso de escucha de canal

**Nota:** Puede utilizar el mandato **kill -9** para finalizar los procesos que no se han detenido.

### Multi **Reinicio de un gestor de colas**

Puede utilizar el mandato **strmqm** para reiniciar un gestor de colas o, en los sistemas Windows y Linux x86-64, puede reiniciar un gestor de colas desde IBM MQ Explorer.

#### **Acerca de esta tarea**

Puede reiniciar un gestor de colas mediante el mandato **strmqm**. Para obtener una descripción del mandato **strmqm** y sus opciones, consulte [strmqm](#).

**Windows** **Linux** En los sistemas Windows y Linux x86-64, puede reiniciar un gestor de colas utilizando IBM MQ Explorer del mismo modo que para iniciar un gestor de colas.

#### **Procedimiento**

- Para reiniciar un gestor de colas mediante el mandato **strmqm**, especifique el mandato seguido del nombre del gestor de colas que desee reiniciar.

Por ejemplo, para iniciar un gestor de colas denominado `strmqm saturn.queue.manager`, especifique el siguiente mandato:

```
strmqm saturn.queue.manager
```

-  

Para iniciar un gestor de colas utilizando IBM MQ Explorer, realice los pasos siguientes:

- a) Abra IBM MQ Explorer.
- b) En la vista de Navegador, seleccione el gestor de colas.
- c) Pulse **Iniciar**.

## Resultados

El gestor de colas se reinicia.

Si el reinicio del gestor de colas tarda más de unos pocos segundos, IBM MQ emite mensajes de información de forma intermitente que detallan el progreso del inicio.

## Visualización y modificación de atributos del gestor de colas

Puede utilizar los mandatos MQSC para visualizar o modificar los atributos del gestor de colas.

### Acerca de esta tarea

Puede utilizar el mandato **DISPLAY QMGR** para visualizar los parámetros de gestor de colas para un gestor de colas y el mandato **ALTER QMGR** para modificar los parámetros de gestor de colas para un gestor de colas local.

### Procedimiento

- Para visualizar los atributos del gestor de colas especificado en el mandato **runmqsc**, utilice el mandato MQSC **DISPLAY QMGR**:

```
DISPLAY QMGR
```

En el ejemplo siguiente se muestra la salida típica de este mandato:

```
DISPLAY QMGR
  1 : DISPLAY QMGR
AMQ8408: Display Queue Manager details.
QMNAME(QM1)
ACCTINT(1800)
ACCTQ(OFF)
ACTVCONO(DISABLED)
ALTDATE(2012-05-27)
AUTHOREV(DISABLED)
CHAD(DISABLED)
CHADEXIT( )
CLWLDATA( )
CLWLLEN(100)
CLWLUSEQ(LOCAL)
CMDLEVEL(800)
CONFIGEV(DISABLED)
CRTIME(16.14.01)
DEFXMITQ( )
DISTL(YES)
IPADDRV(IPV4)
LOGGEREV(DISABLED)
MAXHANDS(256)
MAXPROPL(NOLIMIT)
MAXUMSGS(10000)
MONCHL(OFF)
PARENT( )
PLATFORM(WINDOWSNT)
ACCTCONO(DISABLED)
ACCTMQI(OFF)
ACTIVREC(MSG)
ACTVTRC(OFF)
ALTTIME(16.14.01)
CCSID(850)
CHADEV(DISABLED)
CHLEV(DISABLED)
CLWLXIT( )
CLWLMRUC(999999999)
CMDEV(DISABLED)
COMMANDQ(SYSTEM.ADMIN.COMMAND.QUEUE)
CRDATE(2011-05-27)
DEADQ( )
DESCR( )
INHIBTEV(DISABLED)
LOCALEV(DISABLED)
MARKINT(5000)
MAXMSGL(4194304)
MAXPRTY(9)
MONACLS(QMGR)
MONQ(OFF)
PERFMEV(DISABLED)
PSRTYCNT(5)
```

PSNPMSG (DISCARD)	PSNPRES (NORMAL)
PSSYNCP (IFPER)	QMID (QM1_2011-05-27_16.14.01)
PSMODE (ENABLED)	REMOEEV (DISABLED)
REPOS ( )	REPOSNL ( )
ROUTEREC (MSG)	SCHINIT (QMGR)
SCMDSERV (QMGR)	SSLCRLNL ( )
SSLCRYP ( )	SSLEV (DISABLED)
SSLFIPS (NO)	SSLKEYR (C:\Program Files\IBM\WebSphere
MQ\Data\mqgrs\QM1\ssl\key)	
SSLRKEYC (0)	STATACLS (QMGR)
STATCHL (OFF)	STATINT (1800)
STATMQI (OFF)	STATQ (OFF)
STRSTPEV (ENABLED)	SYNCP
TREELIFE (1800)	TRIGINT (999999999)

**Nota:** SYNCP es un atributo de gestor de colas de sólo lectura.

El parámetro **ALL** es el valor predeterminado en el mandato **DISPLAY QMGR**. Muestra todos los atributos del gestor de colas. En particular, la salida indica el nombre del gestor de colas predeterminado, el nombre de la cola de mensajes no entregados y el nombre de la cola de mandatos.

La existencia de estas colas puede confirmarse entrando el mandato:

```
DISPLAY QUEUE (SYSTEM.*)
```

Esto hace que se visualice una lista de colas que coinciden con la raíz SYSTEM.\*. Los paréntesis son obligatorios.

- Para modificar los atributos del gestor de colas especificado en el mandato **runmqsc**, utilice el mandato **MQSC ALTER QMGR**, especificando los atributos y valores que desee cambiar.

Por ejemplo, utilice los siguientes mandatos para modificar los atributos de `jupiter.queue.manager`:

```
runmqsc jupiter.queue.manager
ALTER QMGR DEADQ (ANOTHERDLQ) INHIBTEV (ENABLED)
```

El mandato **ALTER QMGR** cambia la cola de mensajes no entregados utilizada y habilita los sucesos de inhibición.

Los parámetros no especificados en el mandato **ALTER QMGR** provocan que los valores existentes para esos parámetros se queden sin modificar.

## Tareas relacionadas

[Creación de gestores de colas en Multiplatforms](#)

## Referencia relacionada

[Atributos para el gestor de colas](#)

[runmqsc \(ejecutar mandatos MQSC\)](#)

[DISPLAY QMGR](#)

[ALTER QMGR](#)

## Supresión de un gestor de colas

Puede suprimir un gestor de colas mediante el mandato **dltmqm**. O bien, en los sistemas Windows y Linux, puede utilizar IBM MQ Explorer para suprimir un gestor de colas.


## Antes de empezar



### Atención:



- Suprimir un gestor de colas es una medida muy drástica, ya que también se suprimen todos los recursos asociados al mismo, incluidas todas las colas, sus mensajes y todas las definiciones de objetos. Si utiliza el mandato **dltmqm**, no hay ningún mensaje de solicitud que le permita

cambiar de opinión, por lo tanto, cuando pulse la tecla Intro se perderán todos los recursos asociados.

-  En Windows, al suprimir un gestor de colas también se elimina el gestor de colas de la lista de inicio automático (que se describe en [“Inicio de un gestor de colas” en la página 142](#)). Cuando se haya completado el mandato, se mostrará un mensaje IBM MQ queue manager ending; no se le indicará que el gestor de colas se ha suprimido.
- Al suprimir un gestor de colas de clúster, no se suprime del clúster. Para obtener más información, consulte las notas de uso en [dlmqm](#).

## Acerca de esta tarea

Puede suprimir un gestor de colas mediante el mandato **dlmqm**. Para obtener una descripción del mandato **dlmqm** y sus opciones, consulte [dlmqm](#). Debería asegurarse de que sólo los administradores de confianza tengan autorización para utilizar este mandato. (Para obtener información sobre la seguridad, consulte [Configuración de la seguridad en sistemas UNIX, Linux, and Windows](#).)



  O bien, en los sistemas Windows y Linux (plataformas x86 y x86-64), puede suprimir un gestor de colas utilizando IBM MQ Explorer.

## Procedimiento

- Para suprimir un gestor de colas mediante el mandato **dlmqm**, realice los pasos siguientes:
  - a) Detenga el gestor de colas.
  - b) Emita el mandato siguiente:

```
dlmqm QMB
```

**Nota:** Debe utilizar el mandato **dlmqm** desde la instalación asociada al gestor de colas con el que está trabajando. Puede averiguar con qué instalación está asociado un gestor de colas mediante el mandato `dspmqr -o installation`.

-   Para suprimir un gestor de colas utilizando IBM MQ Explorer, realice los pasos siguientes:
  - a) Abra IBM MQ Explorer.
  - b) En la vista de Navegador, seleccione el gestor de colas.
  - c) Si el gestor de colas no se ha detenido, deténgalo.  
Para detener el gestor de colas, pulse con el botón derecho y seleccione **Detener**.
  - d) Suprima el gestor de colas.  
Para suprimir el gestor de colas, pulse con el botón derecho y seleccione **Suprimir**.

## Resultados

Se suprime el gestor de colas.

## Detención de canales MQI

Cuando se emite un mandato STOP CHANNEL contra un canal de conexión del servidor, puede elegir qué método que se debe utilizar para detener el canal de conexión de cliente. Esto significa que un canal de cliente que emite una llamada de espera MQGET se puede controlar y puede decidir cómo y cuándo se detendrá el canal.

El mandato STOP CHANNEL se puede emitir con tres modalidades, que indican cómo debe detenerse el canal:

### Inmovilizar

Detiene el canal después de que se hayan procesado los mensajes actuales.



Si la compartición de conversaciones está habilitada, el IBM MQ MQI client se percata de la solicitud de detención puntualmente; este tiempo depende de la velocidad de la red. La aplicación cliente se percata de la solicitud de detención como resultado de emitir una llamada subsiguiente a IBM MQ.

#### **Forzar**

Detiene el canal de forma inmediata.

#### **Terminar**

Detiene el canal de forma inmediata. Si el canal se está ejecutando como un proceso, se puede finalizar el proceso de canal o si el canal se está ejecutando como una hebra, finalizará su hebra.

Se trata de un proceso de varias fases. Si se emplea la modalidad de terminación, se intenta detener el canal de conexión con el servidor, primero con la modalidad de desactivación temporal, después con la modalidad de forzar y, si es necesario, con la modalidad de terminación. El cliente puede recibir distintos códigos de retorno durante las distintas fases de terminación. Si se termina el proceso o la hebra, el cliente recibe un error de comunicación.

Los códigos de retorno que se devuelven a la aplicación varían según la llamada MQI emitida y el canal STOP CHANNEL emitido. El cliente recibirá un código de retorno MQRC\_CONNECTION\_QUIESCING o MQRC\_CONNECTION\_BROKEN. Si un cliente detecta MQRC\_CONNECTION\_QUIESCING, debe intentar completar la transacción actual y terminar. Esto no es posible con MQRC\_CONNECTION\_BROKEN. Si el cliente no completa la transacción y termina con suficiente rapidez, obtendrá CONNECTION\_BROKEN al cabo de unos segundos. Es más probable que un mandato STOP CHANNEL con MODE(FORCE) o MODE(TERMINATE) dé como resultado CONNECTION\_BROKEN que con MODE(QUIESCE).

#### **Conceptos relacionados**

[Canales](#)

## **Trabajar con colas locales**

Esta sección contiene ejemplos de algunos mandatos MQSC que puede utilizar para gestionar colas locales, modelo y alias.

Consulte la sección [Mandatos MQSC](#) para obtener información detallada acerca de estos mandatos.

#### **Referencia relacionada**

[Restricciones de nomenclatura para las colas](#)

[Restricciones de nomenclatura para los otros objetos](#)

## **Definición de una cola local con DEFINE QLOCAL**

Para una aplicación, el gestor de colas local es el gestor de colas al que está conectada la aplicación. Las colas que están gestionadas por el gestor de colas local se dice que son locales respecto a ese gestor de colas.

#### **Acerca de esta tarea**

El mandato MQSC **DEFINE QLOCAL** se utiliza para crear una cola local. También puede utilizar el valor predeterminado incluido en la definición de la cola local predeterminada, o puede modificar las características de la cola de las existentes para la cola local predeterminada.

**Nota:** El nombre de la cola local predeterminada es SYSTEM.DEFAULT.LOCAL.QUEUE y se crea durante la instalación del sistema.

#### **Procedimiento**

- Para crear una cola local, especifique el mandato **DEFINE QLOCAL**, tal como se muestra en el ejemplo siguiente.

En este ejemplo, el mandato **DEFINE QLOCAL** define una cola llamada ORANGE.LOCAL.QUEUE con estas características:

- Está habilitada para operaciones get, para operaciones put y funciona de acuerdo con un orden de prioridades.
- Es una cola *normal*, es decir, no es una cola de inicio ni una cola de transmisión y no genera mensajes mensaje desencadenantes.
- La profundidad de cola máxima es 5.000 mensajes; la longitud máxima de un mensaje es de 4.194.304 bytes.

```
DEFINE QLOCAL(ORANGE.LOCAL.QUEUE) +
  DESCR('Queue for messages from other systems') +
  PUT(ENABLED) +
  GET(ENABLED) +
  NOTRIGGER +
  MSGDLVSQ(PRIORITY) +
  MAXDEPTH(5000) +
  MAXMSGL(4194304) +
  USAGE(NORMAL)
```

#### Notas:

1. Salvo el valor para la descripción, todos los valores de atributo que se muestran en el ejemplo son valores predeterminados. Estos ejemplos están incluidos con fines de ilustración. Puede omitirlos si está seguro de que los valores predeterminados son los que desea o no han sido modificados. Consulte también [“Visualización de atributos de objeto predeterminados con DISPLAY QUEUE”](#) en la página 154.
2. **USAGE (NORMAL)** indica que esta cola no es una cola de transmisión.
3. Si ya hay una cola local llamada ORANGE.LOCAL.QUEUE en el mismo gestor de colas, este mandato no podrá ejecutarse. Utilice el atributo **REPLACE** si desea sobrescribir la definición existente de una cola, pero consulte también [“Cambio de los atributos de cola local con ALTER QLOCAL o DEFINE QLOCAL”](#) en la página 156.

#### Referencia relacionada

[DEFINE QLOCAL](#)

## Visualización de atributos de objeto predeterminados con DISPLAY QUEUE

Se puede utilizar el mandato **DISPLAY QUEUE** para visualizar los atributos tomados del objeto predeterminado al definir un objeto IBM MQ.

### Acerca de esta tarea

Cuando define un objeto de IBM MQ, éste toma del objeto predeterminado todos los atributos que no especifique. Por ejemplo, cuando define una cola local, la cola hereda todos los atributos que omite en la definición de la cola local predeterminada, que tiene el nombre SYSTEM.DEFAULT.LOCAL.QUEUE. Puede utilizar el mandato **DISPLAY QUEUE** para ver exactamente cuáles son estos atributos.

### Procedimiento

- Para visualizar los atributos de objeto predeterminados para una cola local, utilice el mandato siguiente:

```
DISPLAY QUEUE (SYSTEM.DEFAULT.LOCAL.QUEUE)
```

La sintaxis del mandato **DISPLAY** es distinta de la del correspondiente mandato **DEFINE**. En el mandato **DISPLAY** basta con dar el nombre de cola, mientras que en el mandato **DEFINE** hay que especificar el tipo de cola, es decir, QLOCAL, QALIAS, QMODEL o QREMOTE.

Puede visualizar de forma selectiva los atributos especificándolos individualmente. Por ejemplo:

```
DISPLAY QUEUE (ORANGE.LOCAL.QUEUE) +
  MAXDEPTH +
```

```
MAXMSGL +  
CURDEPTH;
```

Este mandato muestra los tres atributos especificados de la siguiente manera:

```
AMQ8409: Display Queue details.  
QUEUE(ORANGE.LOCAL.QUEUE) TYPE(QLOCAL)  
CURDEPTH(0) MAXDEPTH(5000)  
MAXMSGL(4194304)
```

**CURDEPTH** es la profundidad de cola actual, es decir, el número de mensajes que hay en la cola. Visualizar este atributo resulta de gran utilidad, ya que puede utilizarlo para supervisar la profundidad de cola y asegurarse de que no se llena.

### Referencia relacionada

[DISPLAY QUEUE](#)

[DEFINE colas](#)

## Copia de una definición de cola local con DEFINE QLOCAL

Puede copiar una definición de cola con el atributo **LIKE** del mandato **DEFINE QLOCAL**.

### Acerca de esta tarea

Puede utilizar el mandato **DEFINE** con el atributo **LIKE** para crear una cola que tiene los mismos atributos que la cola especificada, en vez de tener los atributos de la cola local predeterminada del sistema. También puede utilizar esta forma del mandato **DEFINE** para copiar una definición de cola y sustituir una o más modificaciones en los atributos de la original.

#### Notas:

1. Cuando se utiliza el atributo **LIKE** en un mandato **DEFINE**, solo se copian los atributos de la cola. No se copian los mensajes existente en la cola.
2. Si define una cola local, sin especificar **LIKE**, es lo mismo que:

```
DEFINE LIKE(SYSTEM.DEFAULT.LOCAL.QUEUE)
```

## Procedimiento

- Para crear una cola con los mismos atributos que la cola especificada, en lugar de los de la cola local predeterminada del sistema, especifique el mandato **DEFINE**, tal como se muestra en el ejemplo siguiente.

Escriba el nombre de la cola que se va a copiar exactamente como fue especificada al crearla. Si el nombre contiene caracteres en minúscula, especifique el nombre entre comillas simples.

Este ejemplo crea una cola que tiene los mismos atributos que la cola ORANGE.LOCAL.QUEUE, en vez de tener los atributos de la cola local predeterminada del sistema:

```
DEFINE QLOCAL (MAGENTA.QUEUE) +  
LIKE (ORANGE.LOCAL.QUEUE)
```

- Para copiar una definición de cola, pero sustituir uno o más cambios en los atributos del original, especifique el mandato **DEFINE**, tal como se muestra en el ejemplo siguiente.

Este mandato copia los atributos de la cola ORANGE.LOCAL.QUEUE en la cola THIRD.QUEUE, pero especifica que la longitud máxima de mensaje en la nueva cola será 1024 bytes en lugar de 4194304:

```
DEFINE QLOCAL (THIRD.QUEUE) +  
LIKE (ORANGE.LOCAL.QUEUE) +  
MAXMSGL(1024);
```

## Referencia relacionada

[DEFINE colas](#)

## Cambio de los atributos de cola local con ALTER QLOCAL o DEFINE QLOCAL

Puede cambiar los atributos de cola de dos maneras, utilizando el mandato **ALTER QLOCAL** o el mandato **DEFINE QLOCAL** con el atributo **REPLACE**.

### Acerca de esta tarea

Puede utilizar el atributo **REPLACE** de los mandatos **ALTER** y **DEFINE** para sustituir una definición existente por la definición nueva especificada. Esta diferencia entre utilizar **ALTER** y **DEFINE** radica en que **ALTER** con **REPLACE** no cambia los parámetros no especificados, pero **DEFINE** con **REPLACE** establece todos los parámetros.

### Procedimiento

- Para cambiar los atributos de cola, utilice el mandato **ALTER** o el mandato **DEFINE**, tal como se muestra en los ejemplos siguientes.  
En estos ejemplos, la longitud máxima de mensaje en la cola ORANGE.LOCAL.QUEUE se reduce a 10.000 bytes.

– Utilizando el mandato **ALTER**:

```
ALTER QLOCAL (ORANGE.LOCAL.QUEUE) MAXMSGL(10000)
```

Este mandato modifica un solo atributo, el de la longitud máxima del mensaje; todos los demás atributos no cambian.

– Usando el mandato **DEFINE** con la opción **REPLACE**, por ejemplo: example:

```
DEFINE QLOCAL (ORANGE.LOCAL.QUEUE) MAXMSGL(10000) REPLACE
```

Este mandato no sólo modifica la longitud máxima del mensaje, sino también todos los demás atributos, a los que se les asignan sus valores predeterminados. De modo que, por ejemplo, si anteriormente se inhibió la cola, esto se cambia para habilitarla porque este es el valor predeterminado, como se ha especificado en la cola SYSTEM.DEFAULT.LOCAL.QUEUE.

Si se reduce la longitud máxima del mensaje en una cola existente, los mensajes existentes no se ven afectados. Sin embargo, todos los mensajes nuevos deben cumplir los nuevos criterios.

## Referencia relacionada

[Colas ALTER](#)

[ALTER QLOCAL](#)

[DEFINE colas](#)

[DEFINE QLOCAL](#)

## Borrado de una cola local con CLEAR QLOCAL

Puede utilizar el mandato **CLEAR QLOCAL** para borrar una cola local.

### Antes de empezar

Una cola no se puede vaciar si:

- Hay mensajes no confirmados que se han transferido a la cola bajo punto de sincronización.
- Hay una aplicación que tiene abierta actualmente la cola.

## Acerca de esta tarea

Si desea borrar una cola local utilizando el mandato **CLEAR QLOCAL**, el nombre de la cola debe estar definido en el gestor de colas local.

**Nota:** No hay ningún mensaje de solicitud que le permita cambiar de opinión, por lo tanto, cuando pulse la tecla Intro se perderán los mensajes.

## Procedimiento

Para borrar los mensajes de una cola local, utilice el mandato **CLEAR QLOCAL**, tal como se muestra en el ejemplo siguiente.

En este ejemplo, todos los mensajes se suprimen de una cola local llamada MAGENTA.QUEUE:

```
CLEAR QLOCAL (MAGENTA.QUEUE)
```

### Referencia relacionada

[CLEAR QLOCAL](#)

## Supresión de una cola local con DELETE QLOCAL

Se puede utilizar el mandato MQSC **DELETE QLOCAL** para borrar una cola local.

## Acerca de esta tarea

Una cola no puede suprimirse si contiene mensajes no confirmados.

Si una cola tiene uno o varios mensajes confirmados y ningún mensaje sin confirmar, solo se puede borrar si se especifica la opción **PURGE**. La supresión prosigue incluso si hay mensajes confirmados en la cola indicada y estos mensajes también se depuran.

La especificación de **NOPURGE** en lugar de **PURGE** asegura que la cola no se borre si contiene algún mensaje confirmado.

## Procedimiento

- Para suprimir una cola local, utilice el mandato **DELETE QLOCAL**, tal como se muestra en el ejemplo siguiente.

En este ejemplo se suprime la cola PINK.QUEUE si no hay mensajes confirmados en la cola:

```
DELETE QLOCAL (PINK.QUEUE) NOPURGE
```

En este ejemplo se suprime la cola PINK.QUEUE incluso si hay mensajes confirmados en la cola:

```
DELETE QLOCAL (PINK.QUEUE) PURGE
```

### Referencia relacionada

[DELETE QLOCAL](#)

## Examinar colas con el programa de ejemplo

IBM MQ proporciona un examinador de colas de ejemplo que puede utilizar para examinar el contenido de los mensajes de una cola.

## Acerca de esta tarea

El navegador se proporciona en los formatos fuente y ejecutable en las ubicaciones siguientes, donde *MQ\_INSTALLATION\_PATH* representa el directorio de alto nivel en el que se ha instalado IBM MQ.



```

00000000: 0700 0000 2400 0000 0100 0000 2C00 0000 | .....→..... |
00000010: 0100 0000 0100 0000 0100 0000 AE08 0000 | ..... |
00000020: 0100 0000 0400 0000 4400 0000 DF07 0000 | .....D..... |
00000030: 0000 0000 3000 0000 7361 7475 726E 2E71 | .....0...saturn.q |
00000040: 7565 7565 2E6D 616E 6167 6572 2020 2020 | ueue.manager |
00000050: 2020 2020 2020 2020 2020 2020 2020 2020 | |
00000060: 2020 2020 2020 2020 | |

```

```

No more messages
MQCLOSE
MQDISC

```

## Referencia relacionada

[El programa de ejemplo del navegador](#)

## Habilitar colas grandes

IBM MQ da soporte a colas de más de 2 GB.

**Windows** En sistemas Windows, el soporte para archivos grandes está disponible sin ninguna habilitación adicional.

**Linux** **UNIX** En sistemas UNIX yLinux hay que habilitar explícitamente el soporte de archivos grandes para poder crear archivos de colas de más de 2 GB. Consulte la documentación del sistema operativo para obtener información sobre cómo llevar a cabo esta operación.

Algunos programas de utilidad, por ejemplo tar, no pueden trabajar con archivos de más de 2 GB. Antes de habilitar el soporte de archivos grandes, consulte la documentación del sistema operativo para ver información sobre las restricciones en los programas de utilidad que utilice.

Para obtener información sobre la planificación de la cantidad de almacenamiento necesario para las colas, consulte [Documentos de rendimiento de MQ](#) para informes de rendimiento específicos de plataforma.

**V 9.1.5** A partir de IBM MQ 9.1.5, puede controlar el tamaño de los archivos de cola utilizando un nuevo atributo en las colas locales y de modelo. Consulte [“Modificación de archivos de cola de IBM MQ”](#) en la [página 159](#) para obtener más información.

## **V 9.1.5** **Multi** **Modificación de archivos de cola de IBM MQ**

A partir de IBM MQ 9.1.5, puede controlar el tamaño de los archivos de cola utilizando un nuevo atributo en las colas locales y de modelo. Puede visualizar el tamaño actual de un archivo de cola y el tamaño máximo hasta el que puede crecer actualmente (basándose en el tamaño de bloque en uso actualmente en ese archivo), utilizando dos nuevos atributos de estado de cola.

## Visión general de los nuevos atributos

En colas locales y de modelo, un atributo adicional:

### MAXFSIZE

Denota el tamaño máximo del archivo de cola utilizado por la cola, en megabytes.

Consulte [MAXFSIZE](#) y [“Cambio del tamaño de un archivo de cola de IBM MQ”](#) en la [página 160](#) para obtener más información.

El atributo PCF para este atributo es MQIA\_MAX\_Q\_FILE\_SIZE; consulte [Cambiar, copiar y crear cola](#).

Dos atributos adicionales en el estado de la cola:

### CURFSIZE

Muestra el tamaño actual del archivo de cola en megabytes, redondeado hasta el megabyte más cercano.

Consulte [CURFSIZE](#) para obtener más información.

El atributo PCF para este atributo es MQIA\_CUR\_Q\_FILE\_SIZE.

## CURMAXFS

Indica el tamaño máximo actual hasta el que puede crecer el archivo de cola, redondeado al megabyte más cercano, dado el tamaño de bloque actual en uso en una cola.

Consulte [CURMAXFS](#) para obtener más información.

El atributo PCF para este atributo es MQIA\_CUR\_MAX\_FILE\_SIZE.

Consulte [Consultar cola](#) y [Consultar cola \(respuesta\)](#) para obtener más información sobre estos dos atributos PCF.

Puede establecer y visualizar estos atributos utilizando mandatos MQSC, IBM MQ Explorer y administrative REST API.

**Nota:** solo puede visualizar el valor de MAXFSIZE en la IBM MQ Console; no puede configurar el valor.

## Tamaño de bloque y granularidad

Los archivos de cola se dividen en segmentos denominados bloques. Para aumentar el tamaño máximo de un archivo de cola, es posible que el gestor de colas tenga que cambiar el tamaño de bloque o la granularidad de la cola.

Si se crea una cola recién definida con un valor de MAXFSIZE grande, se crea la cola con un tamaño de bloque adecuado. Sin embargo, si una cola existente tiene su valor de MAXFSIZE aumentado, por ejemplo, utilizando el mandato ALTER QLOCAL, podría ser necesario permitir que la cola se vacíe para que el gestor de colas vuelva a configurar la cola.

Consulte [“Cálculo de la cantidad de datos que un archivo de cola IBM MQ puede almacenar”](#) en la página 161 para obtener más información.



**Atención:** Algunos sistemas de archivos y sistemas operativos tienen límites sobre el tamaño de todo el sistema de archivos o el tamaño de un archivo individual. Debe comprobar los límites en los sistemas que utiliza su empresa.

### Referencia relacionada

[ALTER QUEUES](#)

[DISPLAY QUEUE](#)

[DISPLAY QSTATUS](#)

V 9.1.5

Multi

## **Cambio del tamaño de un archivo de cola de IBM MQ**

Puede aumentar o disminuir el tamaño máximo de un archivo de cola.

## Antes de empezar

Antes de establecer un nuevo tamaño para un archivo de cola, utilice el mandato [DISPLAY QLOCAL](#) para ver el tamaño del archivo de cola que desea cambiar. Por ejemplo, emita el mandato siguiente:

```
DISPLAY QLOCAL(SYSTEM.DEFAULT.LOCAL.QUEUE) MAXFSIZE
```

Recibirá la siguiente salida:

```
AMQ8409I: Display queue details
QUEUE(SYSTEM.DEFAULT.LOCAL.QUEUE)      TYPE(QLOCAL)
MAXFSIZE(DEFAULT)
```

que muestra que el tamaño máximo del archivo de cola es el valor predeterminado de 2.088.960 MB.

## Acerca de esta tarea

En los procedimientos siguientes se muestra cómo:

- Reducir el tamaño máximo hasta el que puede crecer un archivo de cola.
- Aumentar el tamaño máximo hasta el que puede crecer un archivo de cola.





**Atención:** debe tener cuidado al aumentar el tamaño de los archivos de cola sin considerar la forma en que están grabadas las aplicaciones y el posible efecto sobre el rendimiento. El acceso a los mensajes de forma aleatoria en un archivo de cola muy grande puede ser muy lento.

Si está considerando aumentar el tamaño máximo de un archivo de cola más allá del valor predeterminado, debe ser cauteloso al utilizar los selectores de mensajes como, por ejemplo, los ID de correlación y las series de selector de IBM MQ classes for JMS. Los archivos de cola más grandes son más adecuados para el acceso primero en entrar, primero en salir a la cola.

Solo se deben tener cantidades muy grandes de datos en archivos de cola individuales en gestores de colas que están configurados para el registro circular o en los que no se ha habilitado la creación de imágenes de soporte para la cola individual.

No debe limitar el tamaño de las colas SYSTEM, ya que esto podría afectar al funcionamiento del gestor de colas.

## Procedimiento

### 1. Reduzca el tamaño de archivo de cola máximo

- a) Emita el mandato siguiente para crear un archivo local denominado SMALLQUEUE, con un tamaño de 500 gigabytes:

```
DEFINE QLOCAL(SMALLQUEUE) MAXFSIZE(512000)
  2 : DEFINE QLOCAL(SMALLQUEUE) MAXFSIZE(512000)
AMQ8006I: IBM MQ queue created
```

y recibe el mensaje: AMQ8006I:

**Nota:** si configura una cola con un valor menor que la cantidad de datos que ya hay en el archivo, los nuevos mensajes no se podrán poner en la cola.

Si una aplicación intenta poner un mensaje en un archivo de cola que no tiene espacio suficiente, la aplicación recibe el código de retorno MQRC\_Q\_SPACE\_NOT\_AVAILABLE. Cuando se leen suficientes mensajes con lectura destructiva de la cola, las aplicaciones pueden empezar a colocar mensajes nuevos en la cola.

### 2. Aumente el tamaño máximo del archivo de cola.

- a) Emita el mandato siguiente para crear un archivo local denominado LARGEQUEUE, con un tamaño de 5 terabytes:

```
DEFINE QLOCAL(LARGEQUEUE) MAXFSIZE(5242880)
  3 : DEFINE QLOCAL(LARGEQUEUE) MAXFSIZE(5242880)
AMQ8006I: IBM MQ queue created
```

V 9.1.5

Multi

## **Cálculo de la cantidad de datos que un archivo de cola IBM MQ puede almacenar**

La cantidad de datos que se pueden almacenar en una cola está limitada por el tamaño de los bloques individuales en los que se divide la cola.

### Tamaño de bloque y granularidad

El tamaño de bloque predeterminado es 512 bytes. Para admitir archivos de cola mayores de dos terabytes, el gestor de colas tendrá que aumentar el tamaño de bloque.

El tamaño de bloque se calcula automáticamente cuando se configura MAXFSIZE para una cola, pero el tamaño de bloque revisado no se puede aplicar a la cola si la cola ya contiene mensajes. Una vez que una cola está vacía, el gestor de colas modifica automáticamente el tamaño de bloque para admitir el valor de MAXFSIZE configurado.

El mandato DISPLAY QSTATUS tiene un atributo nuevo, CURMAXFS, que le permite confirmar que una cola se ha modificado para utilizar un nuevo tamaño de bloque.

En el ejemplo siguiente, el valor de CURMAXFS de 4177920 confirma que el archivo de cola actualmente puede aumentar hasta aproximadamente cuatro terabytes de tamaño. Si el valor de MAXFSIZE configurado en la cola es mayor que el valor de CURMAXFS, el gestor de colas sigue esperando a que la cola se vacíe antes de que vuelva a configurar el tamaño de bloque del archivo de cola.

```
DISPLAY QSTATUS(LARGEQUEUE) CURMAXFS
  2 : DISPLAY QSTATUS(LARGEQUEUE) CURMAXFS
AMQ8450I: Display queue status details
        QUEUE(LARGEQUEUE)                TYPE(Queue)
        CURMAXFS(4177920)                CURDEPTH(100000)
```

## Comprobación del tamaño de un archivo de cola

Puede visualizar el tamaño actual de un archivo de cola en disco, en megabytes, utilizando el atributo CURFSIZE en el mandato DISPLAY QSTATUS. Esto puede ser útil en plataformas como IBM MQ Appliance, donde no es posible acceder al sistema de archivos directamente.

```
DISPLAY QSTATUS(SMALLQUEUE) CURFSIZE
  1 : DISPLAY QSTATUS(SMALLQUEUE) CURFSIZE
AMQ8450I: Display queue status details
        QUEUE(SMALLQUEUE)                TYPE(Queue)
        CURDEPTH(4024)                   CURFSIZE(10)
```

**Nota:** cuando en una cola se han eliminado mensajes, es posible que el atributo CURFSIZE no disminuya automáticamente.

Normalmente, el espacio no utilizado en un archivo de cola solo se libera en las circunstancias siguientes:

- Cuando ninguna aplicación tiene la cola abierta
- Después de 1000 grabaciones en el registro del gestor de colas, o bien,
- Durante la conclusión del gestor de colas

### Referencia relacionada

[ALTER QUEUES](#)

[DISPLAY QSTATUS](#)

## Trabajar con colas remotas

Una cola remota es una definición local de una cola remota de una cola remota. Es decir, una definición de un gestor de colas local que hace referencia a una cola en un gestor de colas remoto.

No es necesario que defina una cola remota desde una posición local, pero la ventaja de hacerlo es que las aplicaciones pueden hacer referencia a la cola remota por su nombre definido localmente, en lugar de tener que especificar un nombre que esté calificado con el ID del gestor de colas en el que está situado la cola remota.

## Cómo funcionan las definiciones locales de las colas remotas

Una aplicación se conecta a un gestor de colas local y luego emite una llamada MQOPEN. En la llamada de apertura, el nombre de cola especificado es el de una definición de cola remota existente en el gestor de colas local. La definición de cola remota suministra los nombres de la cola de destino, del gestor de colas de destino y, opcionalmente, de una cola de transmisión. Para transferir un mensaje a una cola remota, la aplicación emite una llamada MQPUT especificando el identificador devuelto en la llamada MQOPEN. El gestor de colas utiliza el nombre de la cola remota y el del gestor de colas remoto en una cabecera de transmisión que se añade al principio del mensaje. Esta información se utiliza para dirigir el mensaje a su destino correcto en la red.

Como administrador, puede controlar el destino del mensaje alterando la definición de cola remota.

En el ejemplo siguiente se muestra cómo una aplicación transfiere un mensaje en una cola que es propiedad de un gestor de colas remoto. La aplicación se conecta a un gestor de colas, por ejemplo, saturn.queue.manager. La cola de destino es propiedad de otro gestor de colas.

En la llamada MQOPEN, la aplicación especifica estos campos:

Valor de campo	Descripción
<i>NombreObjeto</i> CYAN.REMOTE.QUEUE	Especifica el nombre local del objeto de cola remota. Esto define la cola de destino y el gestor de colas de destino.
<i>TipoObjeto</i> (Queue)	Identifica este objeto como una cola.
<i>NombGstColasObj</i> En blanco o saturn.queue.manager	Este campo es opcional. Si está en blanco, se presupone el nombre del gestor de colas local. (Es el gestor de colas donde se encuentra la definición de cola remota).

A continuación, la aplicación emite una llamada MQPUT para colocar un mensaje en esta cola.

En el gestor de colas local, puede crear una definición local de una cola remota utilizando los siguientes mandatos MQSC:

```
DEFINE QREMOTE (CYAN.REMOTE.QUEUE) +
DESCR ('Queue for auto insurance requests from the branches') +
RNAME (AUTOMOBILE.INSURANCE.QUOTE.QUEUE) +
RQMNAME (jupiter.queue.manager) +
XMITQ (INQUOTE.XMIT.QUEUE)
```

donde:

#### **QREMOTE (CYAN.REMOTE.QUEUE)**

Especifica el nombre local del objeto de cola remota. Este es el nombre que las aplicaciones conectadas a este gestor de colas deben especificar en la llamada MQOPEN para abrir la cola AUTOMOBILE.INSURANCE.QUOTE.QUEUE del gestor de colas remoto jupiter.queue.manager.

#### **DESCR ('Cola para solicitudes de seguro de coche de las sucursales')**

Proporciona un texto adicional que describe el uso de la cola.

#### **RNAME (AUTOMOBILE.INSURANCE.QUOTE.QUEUE)**

Especifica el nombre de la cola de destino del gestor de colas remoto. Ésta es la cola de destino real para los mensajes enviados por aplicaciones que especifiquen el nombre de cola CYAN.REMOTE.QUEUE. La cola AUTOMOBILE.INSURANCE.QUOTE.QUEUE debe definirse como una cola local en el gestor de colas remoto.

#### **RQMNAME (jupiter.queue.manager)**

Especifica el nombre del gestor de colas remoto al que pertenece la cola de destino AUTOMOBILE.INSURANCE.QUOTE.QUEUE.

#### **XMITQ (INQUOTE.XMIT.QUEUE)**

Especifica el nombre de la cola de transmisión. Es opcional; si no se especifica el nombre de una cola de transmisión, se utiliza una cola con el mismo nombre que el gestor de colas remoto.

En cualquier caso, la cola de transmisión adecuada debe definirse como una cola local con un atributo **Usage** que especifique que es una cola de transmisión (USAGE(XMITQ) en los mandatos MQSC).

## **Forma alternativa de colocar mensajes en una cola remota**

Utilizar una definición local de una cola remota no es la única manera de colocar mensajes en una cola remota. Las aplicaciones pueden especificar el nombre de cola completo, incluido el nombre del gestor de colas remoto, como parte de la llamada MQOPEN. En este caso, no se requiere una definición local de

una cola remota. No obstante, esto significa que las aplicaciones deben conocer o tener acceso al nombre del gestor de colas remoto durante la ejecución.

## Utilización de otros mandatos con colas remotas

Puede utilizar mandatos MQSC para visualizar o alterar los atributos de un objeto cola remota, o puede suprimir el objeto de cola remota. Por ejemplo:

- Para visualizar los atributos de la cola remota:

```
DISPLAY QUEUE (CYAN.REMOTE.QUEUE)
```

- Para cambiar la cola remota para habilitar las transferencias. Esto no afecta a la cola de destino, sólo a las aplicaciones que especifiquen esta cola remota:

```
ALTER QREMOTE (CYAN.REMOTE.QUEUE) PUT(ENABLED)
```

- Para suprimir esta cola remota. Esto no afecta a la cola de destino, sólo a su definición local:

```
DELETE QREMOTE (CYAN.REMOTE.QUEUE)
```

**Nota:** Cuando se suprime una cola remota, sólo se suprime la representación local de la cola remota. No se suprime la cola remota en sí ni ningún mensaje que se encuentre en ella.

## Utilización de definiciones de colas remotas como alias

Además de localizar una cola en otro gestor de colas, también puede utilizar una definición local de una cola remota para los alias de gestor de colas y los alias de cola de respuesta. Ambos tipos de alias se resuelven mediante la definición local de una cola remota. Debe definir los canales adecuados para que el mensaje llegue a su destino.

### Alias de gestor de colas

Un alias es el proceso por el cual el nombre del gestor de colas de destino, tal como se especifica en un mensaje, es modificado por un gestor de colas en la ruta del mensaje. Los alias de gestor de colas son importantes porque se pueden utilizar para controlar el destino de los mensajes dentro de una red de gestores de colas.

Para hacer esto debe alterar la definición de cola remota del gestor de colas en el punto de control. La aplicación emisora no sabe que el nombre de gestor de colas especificado es un alias.

Para obtener más información sobre los alias de gestor de colas, consulte [¿Qué son los alias?](#).

### Alias de cola de respuesta

De forma opcional, una aplicación puede especificar el nombre de una cola de respuesta cuando coloca un *mensaje de solicitud* en una cola.

Si la aplicación que procesa el mensaje extrae el nombre de la cola de respuesta, sabe dónde enviar el *mensaje de respuesta*, si fuera necesario.

Un alias de cola de respuesta es el proceso por el cual una cola de respuesta, tal como se especifica en un mensaje de solicitud, es modificada por un gestor de colas en la ruta del mensaje. La aplicación emisora no sabe que el nombre de cola de respuesta especificado es un alias.

Un alias de cola de respuesta le permite modificar el nombre de la cola de respuesta y, opcionalmente, de su gestor de colas. Esto, a su vez, le permite controlar qué ruta se utiliza para los mensajes de respuesta.

Para obtener más información sobre mensajes de solicitud, mensajes de respuesta y colas de respuesta, consulte las secciones [Tipos de mensajes](#) y [Cola de respuesta y gestor de colas](#).

Para obtener más información sobre las alias de cola de respuesta, consulte [Alias de cola de respuesta y clústeres](#).

## Trabajar con colas alias

Puede definir una cola alias para hacer referencia indirectamente a otra cola o tema.



**Atención:** Las listas de distribución no admiten el uso de cola alias que apuntan a objetos de tema. Si una cola de alias apunta a un objeto de tema en una lista de distribución, IBM MQ devuelve MQRD\_ALIASES\_BASE\_Q\_TYPE\_ERROR.

La cola a la que una cola alias hace referencia puede ser cualquiera de las siguientes:

- Una cola local (consulte el apartado [“Definición de una cola local con DEFINE QLOCAL”](#) en la [página 153](#)).
- Una definición local de una cola remota (consulte el apartado [“Trabajar con colas remotas”](#) en la [página 162](#)).
- Un tema.

Una cola alias no es una cola real, sino una definición que se resuelve en una cola real (o de destino) en tiempo de ejecución. La definición de la cola alias especifica la cola de destino. Cuando una aplicación efectúa una llamada MQOPEN a una cola alias, el gestor de colas resuelve el alias en el nombre de la cola de destino.

Una cola alias no se puede resolver en otra cola alias definida localmente. Sin embargo, una cola alias puede resolverse como colas alias definidas en otro lugar de los clústeres de los cuales forme parte el gestor de colas local. En [Resolución de nombres](#) hallará más información.

Las colas alias son útiles para:

- Proporcionar a las aplicaciones niveles diferentes de autorizaciones de acceso a la cola de destino.
- Permitir que diferentes aplicaciones trabajen con la misma cola de diferentes modos. (Quizá desee asignar prioridades predeterminadas diferentes o valores de persistencia predeterminados diferentes).
- Simplificar el mantenimiento, la migración y el equilibrio de carga. (Quizá desee cambiar el nombre de la cola de destino sin tener que cambiar la aplicación, que sigue utilizando el alias).

Por ejemplo, suponga que se ha desarrollado una aplicación para colocar mensajes en una cola denominada MY.ALIAS.QUEUE. La aplicación especifica el nombre de esta cola cuando realiza una solicitud MQOPEN e, indirectamente, si transfiere un mensaje a esta cola. La aplicación no sabe que la cola es una cola alias. Para cada llamada MQI que utilice este alias, el gestor de colas determina el nombre de cola real, que podría ser una cola local o una cola remota definida en este gestor de colas.

Al cambiar el valor del atributo TARGET, puede redirigir las llamadas MQI a otra cola, posiblemente en otro gestor de colas. Esto resulta de gran utilidad para el mantenimiento, la migración y el equilibrio de la carga.

### Definición de una cola alias

El mandato siguiente crea una cola alias:

```
DEFINE QALIAS (MY.ALIAS.QUEUE) TARGET (YELLOW.QUEUE)
```

Este mandato dirige las llamadas MQI que especifican MY.ALIAS.QUEUE a la cola YELLOW.QUEUE. Este mandato no crea la cola de destino; las llamadas MQI fracasan si la cola YELLOW.QUEUE no existe en tiempo de ejecución.

Si cambia la definición del alias, puede redirigir las llamadas MQI a otra cola. Por ejemplo:

```
ALTER QALIAS (MY.ALIAS.QUEUE) TARGET (MAGENTA.QUEUE)
```

Este mandato dirige las llamadas MQI a otra cola, MAGENTA.QUEUE.

También puede utilizar colas alias para que hacer que una sola cola (la cola destino) parezca tener atributos distintos para aplicaciones distintas. Esto se hace definiendo dos alias, uno para cada aplicación. Suponga que tiene dos aplicaciones:

- La aplicación ALPHA puede transferir mensajes a YELLOW.QUEUE, pero no tiene autorización para obtener mensajes de dicha cola.
- La aplicación BETA puede obtener mensajes de YELLOW.QUEUE, pero no tiene autorización para colocar mensajes en ella.

El mandato siguiente define un alias que está habilitado para operaciones de transferencia e inhabilitado para operaciones de obtención para la aplicación ALPHA:

```
DEFINE QALIAS (ALPHAS.ALIAS.QUEUE) +  
TARGET (YELLOW.QUEUE) +  
PUT (ENABLED) +  
GET (DISABLED)
```

El mandato siguiente define un alias que está inhabilitado para operaciones de transferencia y habilitado para operaciones de obtención para la aplicación BETA:

```
DEFINE QALIAS (BETAS.ALIAS.QUEUE) +  
TARGET (YELLOW.QUEUE) +  
PUT (DISABLED) +  
GET (ENABLED)
```

ALPHA utiliza el nombre de cola ALPHAS.ALIAS.QUEUE en sus llamadas MQI; BETA utiliza el nombre de cola BETAS.ALIAS.QUEUE. Ambas aplicaciones acceden a la misma cola, pero de forma diferente.

Puede utilizar los atributos LIKE y REPLACE cuando defina alias de colas, del mismo modo que los utiliza con colas locales.

## Utilización de otros mandatos con colas alias

Puede utilizar los mandatos MQSC adecuados para visualizar o alterar atributos de la cola alias o para suprimir el objeto de cola alias. Por ejemplo:

Utilice el mandato **DISPLAY QALIAS** para visualizar los atributos de la cola alias:

```
DISPLAY QALIAS (ALPHAS.ALIAS.QUEUE)
```

Utilice el mandato **ALTER QALIAS** para modificar el nombre de cola base al que se resuelve el alias, donde la opción **force** fuerza el cambio incluso si la cola está abierta:

```
ALTER QALIAS (ALPHAS.ALIAS.QUEUE) TARGET(ORANGE.LOCAL.QUEUE) FORCE
```

Use el mandato **DELETE QALIAS** para borrar este alias de cola:

```
DELETE QALIAS (ALPHAS.ALIAS.QUEUE)
```

No se puede suprimir una cola alias si una aplicación tiene actualmente abierta la cola.

### Conceptos relacionados

[Listas de distribución](#)

### Referencia relacionada

[ALTER QALIAS](#)

[DEFINE QALIAS](#)

[DELETE QALIAS](#)

## Trabajar con colas modelo

Un gestor de colas crea una *cola dinámica* si recibe una llamada MQI de una aplicación que especifica un nombre de cola que se ha definido como cola modelo. El nombre de la nueva cola dinámica lo genera el gestor de colas cuando se crea la cola. Una *cola modelo* es una plantilla que especifica los atributos de las colas dinámicas creadas a partir de ella. Las colas modelo proporcionan un método práctico para que las aplicaciones puedan crear colas cuando se necesitan.

### Definición de una cola modelo

Se usa el mandato **DEFINE QMODEL** par definir una colas modelo con un conjunto de atributos del mismo modo en que se define una cola local. Las colas modelo y las colas locales tienen el mismo conjunto de atributos, excepto que en las colas modelo se puede especificar si las colas dinámicas creadas son temporales o persistentes. (Las colas persistentes se conservan tras reiniciar el gestor de colas, las colas temporales no). Por ejemplo:

```
DEFINE QMODEL (GREEN.MODEL.QUEUE) +
DESCR('Queue for messages from application X') +
PUT (DISABLED) +
GET (ENABLED) +
NOTRIGGER +
MSGDLVSQ (FIFO) +
MAXDEPTH (1000) +
MAXMSGL (2000) +
USAGE (NORMAL) +
DEFTYPE (PERMDYN)
```

Este mandato crea una definición de cola modelo. Por el atributo **DEFTYPE** se puede ver que las colas creadas a partir de esta plantilla son colas dinámicas permanentes. Todos los atributos no especificados se copian automáticamente de la cola predeterminada SYSTEM.DEFAULT.MODEL.QUEUE.

Se pueden usar los atributos **LIKE** y **REPLACE** cuando se definen colas modelo de la misma forma en que se usan con las colas locales.

### Utilización de otros mandatos con colas modelo

Puede utilizar los mandatos MQSC adecuados para visualizar o alterar los atributos de una cola modelo o para suprimir el objeto de cola modelo. Por ejemplo:

Use el mandato **DISPLAY QUEUE** para visualizar los atributos de la cola modelo:

```
DISPLAY QUEUE (GREEN.MODEL.QUEUE)
```

Use el mandato **ALTER QMODEL** para modificar el modelo a fin de permitir colocaciones en cualquier cola dinámica creada a partir de dicho modelo:

```
ALTER QMODEL (BLUE.MODEL.QUEUE) PUT(ENABLED)
```

Use el mandato **DELETE QMODEL** para borrar esta cola modelo:

```
DELETE QMODEL (RED.MODEL.QUEUE)
```

### Referencia relacionada

[ALTER QMODEL](#)  
[DEFINE QMODEL](#)  
[DELETE QMODEL](#)  
[DISPLAY QUEUE](#)

## Trabajar con colas de mensajes no entregados

Normalmente, cada gestor de colas tiene una cola local que utilizará como cola de mensajes no entregados, de modo que los mensajes que no se puedan entregar en su destino correcto se almacenen para su recuperación posterior. Debe informar al gestor de colas sobre la cola de mensajes no entregados y especificar cómo se han de procesar los mensajes encontrados en la cola de mensajes no entregados. La utilización de colas de mensajes no entregados puede afectar la secuencia en que se entregan los mensajes, por lo que puede optar por no utilizarlas.

Para informar al gestor de colas sobre la cola de mensajes no entregados, especifique un nombre de cola de mensajes no entregados en el mandato `crtmqm` (`crtmqm -u DEAD.LETTER.QUEUE`, por ejemplo), o utilizando el atributo **DEADQ** en el mandato **ALTER QMGR** para especificar uno más adelante. Debe definir la cola de mensajes no entregados antes de poder utilizarla.

Con el producto se entrega una cola de mensajes no entregados de ejemplo llamada `SYSTEM.DEAD.LETTER.QUEUE`. Esta cola se crea automáticamente cuando se crea el gestor de colas. Si es necesario, puede modificar esta definición y cambiar su nombre.

Una cola de mensajes no entregados no tiene ningún requisito especial excepto que:

- Debe ser una cola local
- Su atributo `MAXMSGL` (longitud máxima de mensajes) debe permitir que la cola pueda alojar los mensajes más grandes que el gestor de colas tenga que manejar **más** el tamaño de la cabecera de mensaje no entregado (`MQDLH`)

La utilización de colas de mensajes no entregados puede afectar la secuencia en que se entregan los mensajes, por lo que puede optar por no utilizarlas. Puede establecer el atributo de canal `USEDLQ` para determinar si la cola de mensajes no entregados se utiliza cuando los mensajes no se pueden entregar. Este atributo puede configurarse de modo que algunas funciones del gestor de colas utilicen la cola de mensajes no entregados mientras que otras no lo hagan. Para obtener más información sobre el uso del atributo de canal `USEDLQ` en mandatos `MQSC` diferentes, consulte [DEFINE CHANNEL](#), [DISPLAY CHANNEL](#), [ALTER CHANNEL](#) y [DISPLAY CLUSQMGR](#).

IBM MQ proporciona un manejador de la cola de mensajes no entregados que le permite especificar cómo se han de procesar o eliminar los mensajes encontrados en una cola de mensajes no entregados. Consulte [“Proceso de mensajes en una cola de mensajes no entregados de IBM MQ”](#) en la página 168.

### Conceptos relacionados

[Colas de mensajes no entregados](#)

### Tareas relacionadas

[Resolución de problemas de mensajes no entregados](#)

### Referencia relacionada

[ALTER QMGR](#)

[crtmqm \(crear gestor de colas\)](#)

## Proceso de mensajes en una cola de mensajes no entregados de IBM MQ

Para procesar mensajes en una cola de mensajes no entregados (DLQ), IBM MQ proporciona un manejador DLQ predeterminado. El manejador DLQ compara los mensajes de la DLQ con las entradas de la tabla de reglas que defina.

Los gestores de colas, los agentes de canales de mensajes (MCA) y las aplicaciones pueden transferir mensajes a una DLQ. Todos los mensajes de la DLQ deben tener como prefijo una estructura de *cabecera de mensaje no entregado*, `MQDLH`. Los mensajes transferidos a la DLQ por un gestor de colas o un agente de canales de mensajes tienen siempre esta cabecera; las aplicaciones que transfieran mensajes a la DLQ deben facilitar esta cabecera. El campo *Reason* de la estructura `MQDLH` contiene un código de razón que indica el motivo por el cual el mensaje está en la DLQ.

Todos los entornos IBM MQ necesitan una rutina para procesar mensajes en la DLQ con regularidad. IBM MQ proporciona una rutina predeterminada, denominada el *manejador de la cola de mensajes no entregados* (el manejador DLQ), que se invoca mediante el mandato `runmqdlq`.



Las instrucciones para procesar mensajes de la DLQ se facilitan al manejador DLQ mediante una *tabla de normas* escrita por el usuario. Es decir, el manejador DLQ compara los mensajes de la DLQ con las entradas de la tabla de reglas; cuando un mensaje DLQ coincide con una de las entradas de la tabla de reglas, el manejador DLQ realiza la acción asociada a dicha entrada.

### Conceptos relacionados

[Colas de mensajes no entregados](#)

### Tareas relacionadas

[Resolución de problemas de mensajes no entregados](#)

## **El manejador de la cola de mensajes no entregados en IBM i**

Qué es el manejador de la cola de mensajes no entregados de IBM i y cómo invocarlo

Una *cola de mensajes no entregados* (DLQ), denominada a veces *cola de mensajes sin entrega* es una cola que retiene los mensajes que no se pueden entregar a las colas de destino. Cada gestor de colas de una red debe tener una DLQ asociada a él.

**Nota:** A menudo es preferible evitar poner mensajes en una DLQ. Para obtener información sobre cuándo utilizar las DLQ y cuándo evitarlas, consulte [“Trabajar con colas de mensajes no entregados”](#) en la página 168.

Los gestores de colas, los agentes de canal de mensajes y las aplicaciones pueden poner mensajes en la DLQ. Todos los mensajes de la DLQ deben tener como prefijo una estructura de *cabecera de mensaje no entregado*, MQDLH. Los mensajes transferidos a la DLQ por un gestor de colas o por un agente de canal de mensajes tienen siempre una MQDLH. Suministre siempre una MQDLH a las aplicaciones que ponen mensajes en la DLQ. El campo *Reason* de la estructura MQDLH contiene un código de razón que indica el motivo por el cual el mensaje está en la DLQ.

En todos los entornos de IBM MQ, debe haber una rutina que se ejecute periódicamente para procesar los mensajes de la DLQ. IBM MQ proporciona una rutina predeterminada, denominada el *manejador de colas de mensajes no entregados* (manejador DLQ), que se invoca mediante el mandato STRMQMDLQ. Una *tabla de reglas* escrita por el usuario proporciona instrucciones al manejador DLQ para procesar los mensajes que hay en la DLQ. Es decir, el manejador DLQ compara los mensajes de la DLQ con las entradas de la tabla de reglas. Cuando un mensaje DLQ coincide con una entrada de la tabla de reglas, el manejador DLQ realiza la acción asociada a dicha entrada.

## Invocación del manejador DLQ

Utilice el mandato STRMQMDLQ para invocar el manejador DLQ. El nombre de la DLQ que desea procesar y del gestor de colas que va a utilizar puede indicarse de dos formas:

- Como parámetros de STRMQMDLQ en la solicitud de mandato. Por ejemplo:

```
STRMQMDLQ UDLMSGQ(ABC1.DEAD.LETTER.QUEUE) SRCMBR(QRULE) SRCFILE(library/QTXTSRC)
MQMNAME(MY.QUEUE.MANAGER)
```

- En la tabla de reglas. Por ejemplo:

```
INPUTQ(ABC1.DEAD.LETTER.QUEUE)
```

**Nota:** La tabla de reglas es un miembro dentro de un archivo físico fuente que puede adoptar cualquier nombre.

Los ejemplos se aplican a la DLQ denominada ABC1.DEAD.LETTER.QUEUE, propiedad del gestor de colas predeterminado.

Si no especifica la DLQ o el gestor de colas tal como se indica, se utilizará el gestor de colas predeterminado de la instalación junto con la DLQ del mismo.

El mandato STRMQMDLQ toma su entrada de la tabla de reglas.

Para poder ejecutar el manejador DLQ debe estar autorizado para acceder a la DLQ y a las colas de mensajes a las que se reenvían los mensajes de la DLQ. También debe estar autorizado para asumir la identidad de otros usuarios, para que la DLQ pueda colocar mensajes en colas con la autorización del ID de usuario del contexto del mensaje.

### Conceptos relacionados

[Colas de mensajes no entregados](#)

### Tareas relacionadas

[Resolución de problemas de mensajes no entregados](#)

#### La tabla de reglas del manejador DLQ en IBM i

La tabla de reglas del manejador de la cola de mensajes no entregados define cómo el manejador DLQ procesa los mensajes que llegan a la cola de mensajes no entregados de IBM i.

La tabla de reglas del manejador DLQ define de qué forma debe el manejador DLQ procesar los mensajes que llegan a la DLQ. Una tabla de reglas tiene dos tipos de entradas:

- La primera entrada de la tabla, que es opcional, contiene *datos de control*.
- Todas las demás entradas de la tabla son *reglas* que el manejador DLQ debe seguir. Cada regla está formada por un *patrón* (un conjunto de características de mensaje) con el que se compara el mensaje, y por una *acción* que debe realizarse cuando un mensaje de la DLQ coincide con el patrón especificado. En una tabla de reglas debe haber una regla como mínimo.

Cada entrada de la tabla de reglas tiene una o más palabras clave.

### Datos de control

Este apartado describe las palabras clave que pueden incluirse en una entrada de datos de control de una tabla de reglas del manejador DLQ. Tenga en cuenta lo siguiente:

- El valor predeterminado de una palabra clave, si lo hay, está subrayado.
- La línea vertical (|) separa alternativas. Sólo puede especificar una de ellas.
- Todas las palabras clave son opcionales.

#### **INPUTQ ( *QueueName* | ' ' (valor predeterminado) )**

El nombre de la DLQ que desea procesar:

1. Cualquier valor UDLMMSGQ (o \*dft) que especifique como parámetro en el mandato **STRMQMDLQ** altera temporalmente cualquier valor INPUTQ en la tabla de reglas.
2. Si especifica un valor UDLMMSGQ en blanco como parámetro del mandato **STRMQMDLQ**, se utiliza el valor INPUTQ de la tabla de reglas.
3. Si especifica un valor UDLMMSGQ en blanco como parámetro del mandato **STRMQMDLQ** y un valor INPUTQ en blanco en la tabla de reglas, se utiliza la cola de mensajes no entregados predeterminada del sistema.

#### **INPUTQM ( *QueueManagerNombre* | ' ' (valor predeterminado) )**

El nombre del gestor de colas que es propietario de la DLQ nombrada en la palabra clave INPUTQ.

Si no especifica ningún gestor de colas o si especifica INPUTQM(' ') en la tabla de reglas, el sistema utiliza el gestor de colas predeterminado de la instalación.

#### **RETRYINT ( *Intervalo* | 60 (valor predeterminado) )**

El intervalo, en segundos, tras el que el manejador DLQ debe intentar volver a procesar mensajes de la DLQ que no se pudieron procesar en el primer intento y de los cuales se han solicitado varios intentos. De forma predeterminada, el intervalo de reintento es de 60 segundos.

#### **WAIT ( YES (valor predeterminado) | NO | *nnn* )**

Indica si el manejador DLQ debe esperar a que lleguen más mensajes a la DLQ cuando detecta que no quedan más mensajes que pueda procesar.

**SI**

Hace que el manejador DLQ espere indefinidamente

**NEE**

Hace que el manejador DLQ termine en cuanto detecta que la DLQ está vacía o no contiene mensajes que pueda procesar.

**nnn**

Hace que el manejador DLQ, después de haber detectado que la cola está vacía o no contiene mensajes que pueda procesar, espere *nnn* segundos a que llegue un nuevo trabajo antes de terminar.

Especifique WAIT (YES) en DLQ ocupadas y WAIT (NO) o WAIT (*nnn*) para las DLQ que tengan un nivel de actividad bajo. Si el manejador DLQ tiene autorización para terminar, invóquelo de nuevo utilizando el mecanismo de desencadenamiento.

Como alternativa a incluir datos de control en la tabla de reglas, puede suministrar el nombre de la DLQ como parámetro de entrada del mandato **STRMQMDLQ**. Si hay un valor que esté especificado tanto en la tabla de reglas como en la entrada del mandato **STRMQMDLQ**, el valor que prevalece es el especificado en el mandato **STRMQMDLQ**.

**Nota:** Si se incluye una entrada de datos de control en la tabla de reglas, debe ser la primera entrada de la tabla.

**IBM i***Reglas de la cola de mensajes no entregados (patrones y acciones) en IBM i*

Una descripción de los patrones y acciones para cada una de las reglas de cola de mensajes no entregados de IBM i.

La siguiente es una regla de ejemplo de una tabla de reglas del manejador DLQ:

```
PERSIST(MQPER_PERSISTENT) REASON (MQRC_PUT_INHIBITED) +
ACTION (RETRY) RETRY (3)
```

Esta regla indica al manejador DLQ que haga 3 intentos de entregar en la cola destino cualquier mensaje persistente que se haya puesto en la DLQ debido a que MQPUT y MQPUT1 estaban inhibidos.

Este apartado describe las palabras clave que pueden incluirse en una regla. Tenga en cuenta lo siguiente:

- El valor predeterminado de una palabra clave, si lo hay, está subrayado. Para la mayoría de palabras clave, el valor predeterminado es \* (asterisco), que coincide con cualquier valor.
- La línea vertical (|) separa alternativas. Sólo puede especificar una de ellas.
- Todas las palabras clave excepto ACTION son opcionales.

Este apartado empieza con una descripción de las palabras clave de coincidencia de patrón (aquellas con las que se comparan los mensajes de la DLQ). A continuación, se describen las palabras clave de acción (aquellas que determinan de qué forma va a procesar el manejador DLQ el mensaje que coincide).

**IBM i***Palabras clave de coincidencia de patrón de DLQ en IBM i*

Las palabras clave de coincidencia de patrón se describen en un ejemplo. Utilice estas palabras clave para especificar valores con los que se comparan los mensajes en la cola de mensajes no entregados de IBM i. Todas las palabras clave de coincidencia con un patrón son opcionales.

**APPLIDAT ( *ApplIdentityDatos*|\* (valor predeterminado))**

El valor *DatosIdentidadAplicación* del mensaje que está en la DLQ especificada en el descriptor de mensaje, MQMD.

**APPLNAME ( *PutApplNombre*|\* (valor predeterminado))**

El nombre de la aplicación que emitió la llamada MQPUT o MQPUT1, según se indica en el campo *NombApplColoc* del descriptor de mensaje, MQMD, del mensaje que está en la DLQ.

**APPLTYPE ( PutApplTipo|\* (valor predeterminado))**

El valor *TipoAplColoc* especificado en el descriptor de mensaje, MQMD, del mensaje que está en la DLQ.

**DESTQ ( QueueName|\* (valor predeterminado))**

El nombre de la cola de mensajes a la que está destinado el mensaje.

**DESTQM ( QueueManagerNombre|\* (valor predeterminado))**

El nombre del gestor de colas de la cola de mensajes a la que está destinado el mensaje.

**FEEDBACK ( Comentarios|\* (valor predeterminado))**

Cuando el valor de *TipoMensaje* es MQMT\_REPORT, *InformaciónRetorno* describe la naturaleza del informe.

Puede utilizar nombres simbólicos. Por ejemplo, puede utilizar el nombre simbólico MQFB\_COA para identificar los mensajes de la DLQ que requieren confirmación de que han llegado a sus respectivas colas de destino.

**FORMAT ( Format|\* (valor predeterminado))**

El nombre que el remitente del mensaje utiliza para describir el formato de los datos del mensaje.

**MSGTYPE ( MsgType|\* (valor predeterminado))**

El tipo de mensaje del mensaje de la DLQ.

Puede utilizar nombres simbólicos. Por ejemplo, puede utilizar el nombre simbólico MQMT\_REQUEST para identificar los mensajes de la DLQ que requieren respuesta.

**PERSIST ( Persistencia|\* (valor predeterminado))**

El valor de persistencia del mensaje. (La persistencia de un mensaje determina si sigue existiendo después de reiniciar el gestor de colas).

Puede utilizar nombres simbólicos. Por ejemplo, puede utilizar el nombre simbólico MQPER\_PERSISTENT para identificar los mensajes de la DLQ que son persistentes.

**REASON ( ReasonCode|\* (valor predeterminado))**

El código de razón que explica por qué se ha colocado el mensaje en la DLQ.

Puede utilizar nombres simbólicos. Por ejemplo, puede utilizar el nombre simbólico MQRC\_Q\_FULL para identificar los mensajes que se colocaron en la DLQ debido a que las colas de destino correspondientes estaban llenas.

**REPLYQ ( QueueName|\* (valor predeterminado))**

El nombre de la cola de respuesta especificado en el descriptor de mensaje, MQMD, del mensaje que está en la DLQ.

**REPLYQM ( QueueManagerNombre|\* (valor predeterminado))**

El nombre del gestor de colas de la cola de respuesta especificada en la palabra clave REPLYQ.

**USERID ( UserIdentifier|\* (valor predeterminado))**

El ID del usuario que originó el mensaje que está en la DLQ, tal como se especifica en el descriptor de mensaje, MQMD.

 **Palabras clave de acción DLQ en IBM i**

Utilice estas palabras clave de acción de la cola de mensajes no entregados para determinar cómo se procesa un mensaje coincidente en la cola de mensajes no entregados de IBM i.

**ACTION (DISCARD|IGNORE|RETRY|FWD)**

La acción que debe realizarse para los mensajes de la DLQ que coincidan con el patrón definido en esta regla.

**DISCARD**

Hace que el mensaje se suprima de la DLQ.

**IGNORE**

Hace que el mensaje se mantenga en la cola de mensajes no entregados (DLQ).

**RETRY**

Hace que el manejador DLQ intente de nuevo poner el mensaje en la cola destino.

**FWD**

Hace que el manejador DLQ reenvíe el mensaje a la cola nombrada en la palabra clave FWDQ.

Debe especificar la palabra clave ACTION. El número de intentos efectuados para llevar a cabo una acción está regido por la palabra clave RETRY. La palabra clave RETRYINT de los datos de control controla el intervalo entre intentos.

**FWDQ ( *QueueName* | &DESTQ | &REPLYQ )**

El nombre de la cola de mensajes a la que se reenvía el mensaje al seleccionar la palabra clave ACTION.

***QueueName***

El nombre de una cola de mensajes. FWDQ(' ') no es válido.

**&DESTQ**

Hace que el nombre de la cola se tome del campo *DestQName* de la estructura MQDLH.

**&REPLYQ**

Toma el nombre de cola del campo *ReplyToQ* del descriptor MQMD del mensaje.

Puede especificar REPLYQ (? \*) en el patrón de mensaje para evitar mensajes de error, cuando una regla que especifica FWDQ (& REPLYQ) coincide con un mensaje con un campo *ReplyToQ* en blanco.

**FWDQM ( *QueueManagerNombre* | & DESTQM | & REPLYQM | ' ' (valor predeterminado) )**

El gestor de colas de la cola a la que se reenvía un mensaje.

***QueueManagerName***

El nombre del gestor de colas de la cola a la que se reenvía el mensaje al seleccionar la palabra clave ACTION (FWD).

**&DESTQM**

Toma el nombre del gestor de colas del campo *DestQMGrName* de la estructura MQDLH.

**&REPLYQM**

Toma el nombre del gestor de colas del campo *ReplyToQMGr* del descriptor MQMD del mensaje.

''

FWDQM(' '), que es el valor predeterminado, identifica el gestor de colas local.

**HEADER ( YES (valor predeterminado) | NO )**

Indica si la MQDLH debe permanecer en un mensaje para el que se solicita la acción ACTION (FWD). Por omisión, la MQDLH permanece en el mensaje. La palabra clave HEADER no es válida para acciones distintas de FWD.

**PUTAUT ( DEF (valor predeterminado) | CTX )**

La autorización con la que el manejador DLQ debe transferir los mensajes:

**DEF**

Transfiere los mensajes con la autorización del propio manejador DLQ.

**CTX**

Hace que los mensajes se transfieran con la autorización del ID de usuario del contexto del mensaje. Si especifica PUTAUT (CTX), debe poseer autorización para asumir la identidad de otros usuarios.

**RETRY ( *RecuentoReintentos* | 1 (valor predeterminado) )**

El número de veces, dentro del rango de 1 a 999.999.999, que se intenta una acción (en el intervalo especificado en la palabra clave RETRYINT de los datos de control).

**Nota:** El número de intentos realizados por el manejador DLQ para implementar una regla concreta es específico de la instancia actual del manejador DLQ; el número total no se conserva al reiniciar. Si reinicia el manejador DLQ, la cuenta de intentos realizados para aplicar una regla se restablece en cero.

La tabla de reglas de la cola de mensajes no entregados de IBM i debe seguir los siguientes convenios de sintaxis, estructura y contenido.

- Una tabla de reglas debe contener una regla como mínimo.
- Las palabras clave pueden estar en cualquier orden.
- Una palabra clave sólo se puede incluir una vez en cualquier regla.
- Las palabras clave no son sensibles a las mayúsculas y minúsculas
- Una palabra clave y el valor de su parámetro deben estar separados de las demás palabras clave por un blanco o una coma como mínimo.
- Al principio o al final de una regla y entre palabras clave, signos de puntuación y valores, puede haber cualquier número de espacios en blanco.
- Cada regla debe empezar en una nueva línea.
- Por motivos de portabilidad, la longitud significativa de una línea no debe ser superior a los 72 caracteres.
- Utilice el signo más (+) en una línea como último carácter distinto del blanco para indicar que la regla continúa en el primer carácter distinto del blanco de la línea siguiente. Utilice el signo menos (-) en una línea como último carácter distinto del blanco para indicar que la regla continúa en el principio de la línea siguiente. Puede haber caracteres de continuación dentro de palabras clave y parámetros.

Por ejemplo:

```
APPLNAME (' ABC+
D')
```

da como resultado 'ABCD'.

```
APPLNAME (' ABC -
D')
```

da como resultado 'ABC D'.

- Puede haber líneas de comentario, que empiezan por un asterisco (\*), en cualquier lugar de la tabla de reglas.
- Las líneas en blanco se ignoran.
- Cada entrada de la tabla de reglas del manejador DLQ incluye una o más palabras clave y sus parámetros asociados. Los parámetros deben seguir estas reglas sintácticas:
  - Cada valor de parámetro debe contener, como mínimo, un carácter significativo. Las comillas delimitadoras de los valores entre comillas no se consideran significativas. Por ejemplo, serían válidos estos parámetros:

FORMAT (' ABC ')	3 caracteres significativos
FORMAT (ABC)	3 caracteres significativos
FORMAT (' A ')	1 carácter significativo
FORMAT (A)	1 carácter significativo
FORMAT (' ')	1 carácter significativo

Estos parámetros no son válidos porque no contienen caracteres significativos:

```
FORMAT (' ' )
FORMAT ( )
```

FORMAT ( )

FORMAT

- Se admiten caracteres comodín. Puede utilizar el signo de interrogación (?) en lugar de cualquier carácter individual, excepto un blanco final. Puede utilizar el asterisco (\*) en lugar de cero o más caracteres adyacentes. El asterisco (\*) y el signo de interrogación (?) se interpretan **siempre** como caracteres comodín en los valores de parámetros.
- No puede incluir caracteres comodín en los parámetros de estas palabras clave: ACTION, HEADER, RETRY, FWDQ, FWDQM y PUTAUT.
- Los espacios en blanco de cola de los valores de parámetros y de los campos correspondientes en los mensajes de la DLQ no son significativos para realizar comparaciones con comodines. Sin embargo, los espacios en blanco de cabecera e intercalados en series de caracteres entre comillas sí son significativos en las comparaciones con comodines.
- Los parámetros numéricos no pueden incluir el carácter comodín de signo de interrogación (?). Puede incluir un asterisco (\*) para que haga las veces de todo un parámetro numérico, pero el asterisco no puede formar parte del parámetro numérico. Por ejemplo, serían válidos estos parámetro numéricos:

MSGTYPE (2)                      Sólo pueden elegirse mensajes de respuesta

MSGTYPE (\*)                      Puede elegirse cualquier tipo de mensaje

MSGTYPE ( ' \* ' )                Puede elegirse cualquier tipo de mensaje

Sin embargo, MSGTYPE ( ' 2 \* ' ) no es válido, porque incluye un asterisco (\*) como parte de un parámetro numérico.

- Los parámetros numéricos deben estar comprendidos entre 0 y 999.999.999. Si el valor del parámetro está dentro de este rango, se aceptará, incluso si en ese momento no es válido en el campo asociado a la palabra clave. Puede utilizar nombres simbólicos para parámetros numéricos.
- Si un valor de tipo serie es más corto que el campo de MQDLH o MQMD con el que está relacionada la palabra clave, el valor se rellenará con espacios en blanco hasta alcanzar la longitud del campo. Si el valor, excluyendo los asteriscos, es más largo que la longitud del campo, se diagnosticará un error. Por ejemplo, serían válidos todos estos valores de tipo serie para un campo de 8 caracteres:

' ABCDEFGH '                      8 caracteres

' A \* C \* E \* G \* I '                5 caracteres, excluidos los asteriscos

' \* A \* C \* E \* G \* I \* K \* M \* O \* '    8 caracteres, excluidos los asteriscos

- Las series que contienen espacios en blanco, caracteres en minúsculas o caracteres especiales distintos del punto (.), la barra inclinada (?), el subrayado ( \_ ) y el signo de porcentaje (%) deben ir entre comillas simples. Los caracteres en minúsculas que no estén entre comillas se convierten a mayúsculas. Si la serie incluye una parte entre comillas, se deben utilizar dos comillas simples para indicar el principio y el final de la parte entrecomillada. Cuando se calcula la longitud de la serie, cada aparición de comillas dobles se cuenta como un solo carácter.

#### IBM i

#### Proceso de la tabla de reglas de la cola de mensajes no entregados en IBM i

El manejador de la cola de mensajes no entregados busca en la tabla de reglas una regla con un patrón que coincida con un mensaje en la cola de mensajes no entregados de IBM i.

La búsqueda empieza por la primera regla de la tabla y continúa secuencialmente por la tabla. Cuando se encuentra una regla cuyo patrón coincide, la tabla de reglas intenta realizar la acción indicada por la regla. El manejador DLQ aumenta en una unidad la cuenta de reintentos de la regla cada vez que intenta aplicarla. Si falla el primer intento, el intento se repite hasta que la cuenta de intentos sea igual al número especificado en la palabra clave RETRY. Si todos los intentos fallan, el manejador DLQ busca la siguiente regla coincidente de la tabla.

Este proceso se repite para las siguientes reglas coincidentes hasta que se realiza una acción satisfactoriamente. Cuando se han intentado realizar todas las reglas coincidentes el número de veces

especificado en su palabra clave RETRY y todos los intentos han fallado, se presupone ACTION (IGNORE). ACTION (IGNORE) también se presupone si no se encuentra ninguna regla coincidente.

**Nota:**

1. Sólo se buscan patrones de reglas coincidentes para mensajes de la DLQ que empiecen por MQDLH. Los mensajes que no empiezan por MQDLH se indican periódicamente como erróneos, y permanecen en la DLQ indefinidamente.
2. Todas las palabras clave de patrón pueden tomar su valor predeterminado, por lo que una regla puede constar únicamente de una acción. No obstante, tenga en cuenta que las reglas que son sólo una acción se aplican a todos los mensajes de la cola que tienen MQDLH y que no se han procesado todavía de acuerdo con otras reglas de la tabla.
3. La tabla de reglas se valida al iniciar el manejador DLQ y en ese momento se marcan los errores con un distintivo. (Los mensajes de error que emite el manejador DLQ se describen en el manual [Mensajes y códigos de razón](#)). Puede realizar cambios en la tabla de reglas siempre que lo desee, pero los cambios no entrarán en vigor hasta que reinicie el manejador DLQ.
4. El manejador DLQ no altera el contenido de los mensajes, de la MQDLH ni del descriptor de mensaje. El manejador DLQ siempre pone mensajes en otras colas con la opción de mensaje MQPMO\_PASS\_ALL\_CONTEXT.
5. Es posible que no se reconozcan los errores sintácticos consecutivos de la tabla de reglas debido a que la validación de la tabla de reglas elimina la generación de errores repetitivos.
6. El manejador DLQ abre la DLQ con la opción MQOO\_INPUT\_AS\_Q\_DEF.
7. Pueden ejecutarse simultáneamente varias instancias del manejador DLQ para la misma cola, utilizando la misma tabla de reglas. Pero es más normal que exista una relación unívoca entre una DLQ y un manejador DLQ.

**IBM i**

*Asegurarse de que todos los mensajes de DLQ se procesan en IBM i*

El manejador de la cola de mensajes no entregados mantiene un registro de todos los mensajes en la DLQ de IBM i que se han visto pero no eliminado.

Si se utiliza el manejador DLQ como filtro para extraer un pequeño subconjunto de mensajes de la DLQ, el manejador DLQ sigue llevando un registro de los mensajes de la DLQ que no ha procesado. El manejador DLQ tampoco puede garantizar que se vayan a ver los nuevos mensajes que lleguen a la DLQ, incluso si la DLQ está definida como FIFO (primero en entrar, primero en salir). Si la cola no está vacía, se realizan nuevas exploraciones periódicas de la DLQ para comprobar todos los mensajes.

Por ello, conviene asegurarse de que la DLQ contenga el menor número posible de mensajes. Si a los mensajes que no se pueden descartar ni reenviar a otras colas (sea cual sea el motivo) se les permite acumularse en la cola, la carga de trabajo del manejador DLQ aumenta y la propia DLQ corre el riesgo de llenarse.

Puede tomar medidas específicas que permitan al manejador DLQ vaciar la DLQ. Por ejemplo, intente no utilizar ACTION (IGNORE), que deja los mensajes en la DLQ. (Recuerde que se presupone ACTION (IGNORE) para los mensajes a los que no se aplica explícitamente ninguna otra regla de la tabla). Para estos mensajes que de otro modo se ignorarían, utilice una acción que traslade los mensajes a otra cola. Por ejemplo:

```
ACTION (FWD) FWDQ (IGNORED.DEAD.QUEUE) HEADER (YES)
```

Del mismo modo, haga que la regla final de la tabla procese todos los mensajes en los que no hayan actuado las reglas anteriores de la tabla. Por ejemplo, la regla final de la tabla puede ser parecida a esto:

```
ACTION (FWD) FWDQ (REALLY.DEAD.QUEUE) HEADER (YES)
```

Esto provoca que los mensajes que pasan a la regla final de la tabla se reenvíen a la cola REALLY.DEAD.QUEUE, donde se pueden procesar manualmente. Si no tiene una regla de este tipo, es probable que los mensajes permanezcan en la DLQ indefinidamente.



Ejemplo de código para una tabla de reglas del manejador de la cola de mensajes no entregados en IBM i. Este es un ejemplo de tabla de reglas que contiene una sola entrada de datos de control y varias reglas.

```

*****
*   An example rules table for the STRMQMDLQ command   *
*****
* Control data entry
* -----
* If no queue manager name is supplied as an explicit parameter to
* STRMQMDLQ, use the default queue manager for the machine.
* If no queue name is supplied as an explicit parameter to STRMQMDLQ,
* use the DLQ defined for the local queue manager.
*
inputqm(' ') inputq(' ')

* Rules
* -----
* We include rules with ACTION (RETRY) first to try to
* deliver the message to the intended destination.

* If a message is placed on the DLQ because its destination
* queue is full, attempt to forward the message to its
* destination queue. Make 5 attempts at approximately
* 60-second intervals (the default value for RETRYINT).
REASON(MQRC_Q_FULL) ACTION(RETRY) RETRY(5)

* If a message is placed on the DLQ because of a put inhibited
* condition, attempt to forward the message to its
* destination queue. Make 5 attempts at approximately
* 60-second intervals (the default value for RETRYINT).
REASON(MQRC_PUT_INHIBITED) ACTION(RETRY) RETRY(5)

* The AAAA corporation is always sending messages with incorrect
* addresses. When we find a request from the AAAA corporation,
* we return it to the DLQ (DEADQ) of the reply-to queue manager
* (&REPLYQM).
* The AAAA DLQ handler attempts to redirect the message.
MSGTYPE(MQMT_REQUEST) REPLYQM(AAAA.*) +
ACTION(FWD) FWDQ(DEADQ) FWDQM(&REPLYQM)

* The BBBB corporation never does things by half measures. If
* the queue manager BBBB.1 is unavailable, try to
* send the message to BBBB.2
DESTQM(bbbb.1) +
action(fwd) fwdq(&DESTQ) fwdqm(bbbb.2) header(no)

* The CCCC corporation considers itself very security
* conscious, and believes that none of its messages
* will ever end up on one of our DLQs.
* Whenever we see a message from a CCCC queue manager on our
* DLQ, we send it to a special destination in the CCCC organization
* where the problem is investigated.
REPLYQM(CCCC.*) +
ACTION(FWD) FWDQ(ALARM) FWDQM(CCCC.SYSTEM)

* Messages that are not persistent run the risk of being
* lost when a queue manager terminates. If an application
* is sending nonpersistent messages, it must be able
* to cope with the message being lost, so we can afford to
* discard the message.
PERSIST(MQPER_NOT_PERSISTENT) ACTION(DISCARD)

* For performance and efficiency reasons, we like to keep
* the number of messages on the DLQ small.
* If we receive a message that has not been processed by
* an earlier rule in the table, we assume that it
* requires manual intervention to resolve the problem.
* Some problems are best solved at the node where the
* problem was detected, and others are best solved where

```

```
* the message originated. We do not have the message origin,  
* but we can use the REPLYQM to identify a node that has  
* some interest in this message.  
* Attempt to put the message onto a manual intervention  
* queue at the appropriate node. If this fails,  
* put the message on the manual intervention queue at  
* this node.
```

```
REPLYQM('?*') +  
ACTION(FWD) FWDQ(DEADQ.MANUAL.INTERVENTION) FWDQM(&REPLYQM)
```

```
ACTION(FWD) FWDQ(DEADQ.MANUAL.INTERVENTION)
```

## Invocación del manejador DLQ

Invoke el manejador de la cola de mensajes no entregados mediante el mandato `runmqdlq`. Puede especificar la DLQ que desea procesar y el gestor de colas que desea utilizar de dos formas.

Los dos formas son las siguientes:

- Como parámetros de `runmqdlq` desde el indicador de mandatos. Por ejemplo:

```
runmqdlq ABC1.DEAD.LETTER.QUEUE ABC1.QUEUE.MANAGER <qrule.rul
```

- En la tabla de reglas. Por ejemplo:

```
INPUTQ(ABC1.DEAD.LETTER.QUEUE) INPUTQM(ABC1.QUEUE.MANAGER)
```

Los ejemplos se refieren a la DLQ llamada `ABC1.DEAD.LETTER.QUEUE`, que pertenece al gestor de colas `ABC1.QUEUE.MANAGER`.

Si no especifica la DLQ o el gestor de colas tal como se indica, se utilizará el gestor de colas predeterminado de la instalación junto con la DLQ del mismo.

El mandato `runmqdlq` toma su entrada de `stdin`; la tabla de reglas con `runmqdlq` se asocia redireccionando `stdin` desde la tabla de reglas.

Para ejecutar el manejador DLQ, debe tener autorización de acceso a la DLQ propiamente dicha y a todas las colas de mensajes a las que se envíen los mensajes de la DLQ. Además, para que el manejador DLQ pueda transferir mensajes a colas con la autorización del ID de usuario del contexto del mensaje, debe tener autorización para asumir la identidad de otros usuarios.

Para obtener más información sobre el mandato `runmqdlq`, consulte [runmqdlq](#).

### Conceptos relacionados

[Colas de mensajes no entregados](#)

### Tareas relacionadas

[Resolución de problemas de mensajes no entregados](#)

#### El manejador DLQ de ejemplo `amqsdllq`

Además del manejador de la cola de mensajes no entregados invocado mediante el mandato `runmqdlq`, IBM MQ proporciona el código fuente de un manejador DLQ de ejemplo, `amqsdllq`, con una función que es similar a la que proporciona `runmqdlq`.

Puede personalizar `amqsdllq` para proporcionar un manejador DLQ que satisfaga sus requisitos. Por ejemplo, puede que necesite un manejador DLQ que pueda procesar mensajes sin cabeceras de mensajes no entregados. (Tanto el manejador DLQ predeterminado como el de ejemplo, `amqsdllq`, procesan únicamente los mensajes de la DLQ que empiezan por una cabecera de mensaje no entregado, `MQDLH`. Los mensajes que no empiezan por una `MQDLH` se identifican como erróneos y permanecen indefinidamente en la DLQ).

`MQ_INSTALLATION_PATH` representa el directorio de alto nivel en el que está instalado IBM MQ.

En IBM MQ for Windows, el código fuente de amqsdlq se encuentra en el directorio:

```
MQ_INSTALLATION_PATH\tools\c\samples\dlq
```

y la versión compilada se encuentra en el directorio:

```
MQ_INSTALLATION_PATH\tools\c\samples\bin
```

En IBM MQ for UNIX y sistemas Linux, el código fuente de amqsdlq se encuentra en el directorio:

```
MQ_INSTALLATION_PATH/samp/dlq
```

y la versión compilada se encuentra en el directorio:

```
MQ_INSTALLATION_PATH/samp/bin
```

También puede compilar **amqsdlq** en modalidad de cliente. Para obtener más información, consulte [Escribir aplicaciones de procedimiento de cliente, Creación de aplicaciones para IBM MQ MQI clients y Ejecución de aplicaciones en el entorno de IBM MQ MQI client.](#)

### **La tabla de reglas del manejador DLQ**

La tabla de reglas del manejador de la cola de mensajes no entregados define cómo debe el manejador DLQ procesar los mensajes que llegan a la DLQ.

Una tabla de reglas tiene dos tipos de entradas:

- La primera entrada de la tabla, que es opcional, contiene *datos de control*.
- Todas las demás entradas de la tabla son *reglas* que el manejador DLQ debe seguir. Cada regla está formada por un *patrón* (un conjunto de características de mensaje) con el que se compara el mensaje, y por una *acción* que debe realizarse cuando un mensaje de la DLQ coincide con el patrón especificado. En una tabla de reglas debe haber una regla como mínimo.

Cada entrada de la tabla de reglas tiene una o más palabras clave.

#### **Conceptos relacionados**

[Colas de mensajes no entregados](#)

#### **Tareas relacionadas**

[Resolución de problemas de mensajes no entregados](#)

#### *Datos de control de DLQ*

Puede incluir palabras clave en una entrada de datos de control de una tabla de reglas del manejador de la cola de mensajes no entregados.

#### **Nota:**

- La línea vertical (|) separa alternativas, de las que sólo puede especificarse una.
- Todas las palabras clave son opcionales.

#### **INPUTQ ( *QueueName* | ' ' (valor predeterminado) )**

El nombre de la DLQ que desea procesar:

1. Cualquier valor INPUTQ que especifique como parámetro en el mandato `runmqdlq`, prevalece sobre todos los valores INPUTQ de la tabla de reglas.
2. Si no especifica ningún valor INPUTQ como parámetro en el mandato `runmqdlq`, pero **sí** especifica un valor en la tabla de reglas, se utiliza el valor INPUTQ de la tabla de reglas.
3. Si no se especifica ninguna DLQ o si se especifica INPUTQ(' ') en la tabla de reglas, se utiliza el nombre de la DLQ que pertenece al gestor de colas cuyo nombre se proporciona como parámetro en el mandato `runmqdlq`.

4. Si no especifica ningún valor INPUTQ como parámetro en el mandato runmqdlq o como valor en la tabla de reglas, se utiliza la DLQ que pertenece al gestor de colas nombrado en la palabra clave INPUTQM de la tabla de reglas.

#### **INPUTQM ( *QueueManagerNombre* | ' ' (valor predeterminado))**

El nombre del gestor de colas al que pertenece la DLQ nombrada en la palabra clave INPUTQ:

1. Cualquier valor INPUTQM que especifique como parámetro en el mandato runmqdlq, prevalece sobre todos los valores INPUTQM de la tabla de reglas.
2. Si no especifica ningún valor INPUTQM como parámetro en el mandato runmqdlq, se utiliza el valor INPUTQM de la tabla de reglas.
3. Si no especifica ningún gestor de colas o especifica INPUTQM(' ') en la tabla de reglas, se utiliza el gestor de colas predeterminado de la instalación.

#### **RETRYINT ( *Intervalo* | 60 (valor predeterminado))**

Es el intervalo, en segundos, tras el cual el manejador DLQ debe intentar volver a procesar mensajes de la DLQ que no pudo procesar en el primer intento, y para los cuales se han solicitado varios intentos. De forma predeterminada, el intervalo de reintento es de 60 segundos.

#### **WAIT ( YES (valor predeterminado) | NO | nnn )**

Indica si el manejador DLQ debe esperar a que lleguen más mensajes a la DLQ cuando detecta que no quedan más mensajes que pueda procesar.

##### **SÍ**

Hace que el manejador DLQ espere indefinidamente.

##### **NEE**

Hace que el manejador DLQ termine cuando detecta que la DLQ está vacía o no contiene mensajes que pueda procesar.

##### **nnn**

Hace que el manejador DLQ, después de haber detectado que la cola está vacía o no contiene mensajes que pueda procesar, espere *nnn* segundos a que llegue un nuevo trabajo antes de terminar.

Especifique WAIT (YES) en DLQ ocupadas y WAIT (NO) o WAIT (*nnn*) para las DLQ que tengan un nivel de actividad bajo. Si el manejador DLQ tiene autorización para terminar, es aconsejable invocarlo de nuevo mediante la activación. Para obtener más información sobre el desencadenamiento, consulte [Inicio de aplicaciones IBM MQ utilizando desencadenantes](#).

Como alternativa a la inclusión de datos de control en la tabla de reglas, puede facilitar los nombres de la DLQ y su gestor de colas como parámetros de entrada del mandato runmqdlq. Si se especifican valores tanto en la tabla de reglas como en la entrada del mandato runmqdlq, prevalece el valor especificado en el mandato runmqdlq.

Si incluye una entrada de datos de control en la tabla de reglas, dicha entrada ser la **primera** de la tabla.

#### *Reglas de la cola de mensajes no entregados (patrones y acciones)*

Una descripción de las palabras clave de coincidencia de patrón (aquellas con las que se comparan los mensajes de la cola de mensajes no entregados) y las palabras clave de acción (aquellas que determinan la forma en que el manejador de la cola de mensajes no entregados debe procesar un mensaje coincidente). También se proporciona una regla de ejemplo.

## **Palabras clave de coincidencia de patrón**

Las palabras clave de coincidencia de patrón, que se utilizan para especificar valores con los que se comparan los mensajes de la DLQ, son las siguientes. (Todas las palabras clave de coincidencia de patrón son opcionales):

#### **APPLIDAT ( *ApplIdentityDatos* | \* (valor predeterminado))**

El valor de *ApplIdentityData* indicado en el descriptor del mensaje, MQMD, del mensaje de la DLQ.

**APPLNAME ( *PutApplNombre* | \* (valor predeterminado))**

El nombre de la aplicación que ha emitido las llamadas MQPUT o MQPUT1, tal y como se especifica en el campo *NombApplColoc* del descriptor de mensaje, MQMD, del mensaje en la DLQ.

**APPLTYPE ( *PutApplTipo* | \* (valor predeterminado))**

El valor *TipoApplColoc*, especificado en el descriptor de mensaje, MQMD, del mensaje que está en la DLQ.

**DESTQ ( *QueueName* | \* (valor predeterminado))**

El nombre de la cola de mensajes a la que está destinado el mensaje.

**DESTQM ( *QueueManagerNombre* | \* (valor predeterminado))**

El nombre del gestor de colas de la cola de mensajes a la que está destinado el mensaje.

**FEEDBACK ( *Comentarios* | \* (valor predeterminado))**

Cuando el valor de *TipoMsj* es MQFB\_REPORT, *Respuesta* describe la naturaleza del informe.

Puede utilizar nombres simbólicos. Por ejemplo, puede utilizar el nombre simbólico MQFB\_COA para identificar los mensajes de la DLQ que requieren confirmación de la llegada a sus respectivas colas de destino.

**FORMAT ( *Format* | \* (valor predeterminado))**

El nombre que el remitente del mensaje utiliza para describir el formato de los datos del mensaje.

**MSGTYPE ( *MsgType* | \* (valor predeterminado))**

El tipo de mensaje del mensaje de la DLQ.

Puede utilizar nombres simbólicos. Por ejemplo, puede utilizar el nombre simbólico MQMT\_REQUEST para identificar los mensajes de la DLQ que requieren respuesta.

**PERSIST ( *Persistencia* | \* (valor predeterminado))**

El valor de persistencia del mensaje. (La persistencia de un mensaje determina si sigue existiendo después de reiniciar el gestor de colas).

Puede utilizar nombres simbólicos. Por ejemplo, puede utilizar el nombre simbólico MQPER\_PERSISTENT para identificar mensajes de la DLQ que son persistentes.

**REASON ( *ReasonCode* | \* (valor predeterminado))**

El código de razón que explica por qué se ha colocado el mensaje en la DLQ.

Puede utilizar nombres simbólicos. Por ejemplo, puede utilizar el nombre simbólico MQRC\_Q\_FULL para identificar los mensajes que se colocaron en la DLQ debido a que las colas de destino correspondientes estaban llenas.

**REPLYQ ( *QueueName* | \* (valor predeterminado))**

El nombre de la cola de respuesta especificado en el descriptor del mensaje, MQMD, del mensaje de la DLQ.

**REPLYQM ( *QueueManagerNombre* | \* (valor predeterminado))**

El nombre del gestor de colas de la cola de respuesta especificado en el descriptor de mensaje, MQMD, del mensaje de la DLQ.

**USERID ( *UserIdentifier* | \* (valor predeterminado))**

El ID del usuario que originó el mensaje que está en la DLQ, tal como se especifica en el descriptor de mensaje, MQMD, del mensaje de la DLQ.

## Las palabras clave de acción

Las palabras clave de acción, que se utilizan para describir la forma de procesar un mensaje, son las siguientes:

**ACTION (DISCARD|IGNORE|RETRY|FWD)**

Es la acción que debe realizarse para cualquier mensaje de la DLQ que coincida con el patrón definido en esta regla.

**DISCARD**

Hace que el mensaje se suprima de la DLQ.

**IGNORE**

Hace que el mensaje permanezca en la DLQ.

**RETRY**

Si el primer intento de transferir el mensaje a la cola de destino no se ejecuta correctamente, hace que se vuelva a intentar. La palabra clave RETRY establece el número de intentos realizados para implementar una acción. La palabra clave RETRYINT de los datos de control controla el intervalo entre intentos.

**FWD**

Hace que se remita el mensaje a la cola mencionada en la palabra clave FWDQ.

Debe especificar la palabra clave ACTION.

**FWDQ ( *QueueName* | &DESTQ | &REPLYQ )**

El nombre de la cola de mensajes a la que debe remitirse el mensaje cuando se solicita la acción ACTION (FWD).

***QueueName***

El nombre de una cola de mensajes. FWDQ(' ') no es válido.

**&DESTQ**

Hace que el nombre de la cola se tome del campo *DestQName* de la estructura MQDLH.

**&REPLYQ**

Toma el nombre de cola del campo *ReplyToQ* del descriptor MQMD del mensaje.

Para evitar mensajes de error cuando una regla que especifica FWDQ (& REPLYQ) coincide con un mensaje con un campo *ReplyToQ* en blanco, especifique REPLYQ (? \*) en el patrón del mensaje.

**FWDQM ( *QueueManagerNombre* | & DESTQM | & REPLYQM | ' ' (valor predeterminado) )**

El gestor de colas de la cola a la que debe remitirse un mensaje.

***QueueManagerName***

El nombre del gestor de colas de la cola a la que debe remitirse un mensaje cuando se solicita la acción ACTION (FWD).

**&DESTQM**

Toma el nombre del gestor de colas del campo *DestQMGrName* de la estructura MQDLH.

**&REPLYQM**

Toma el nombre del gestor de colas del campo *ReplyToQMGr* del descriptor MQMD del mensaje.

''

FWDQM(' '), que es el valor predeterminado, identifica el gestor de colas local.

**HEADER ( YES (valor predeterminado) | NO )**

Indica si la MQDLH debe permanecer en un mensaje para el que se solicita la acción ACTION (FWD). Por omisión, la MQDLH permanece en el mensaje. La palabra clave HEADER no es válida para acciones distintas de FWD.

**PUTAUT ( DEF (valor predeterminado) | CTX )**

La autorización con la que el manejador DLQ debe transferir los mensajes:

**DEF**

Hace que los mensajes se transfieran con la autorización del propio manejador DLQ.

**CTX**

Hace que los mensajes se transfieran con la autorización del ID de usuario del contexto del mensaje. Si especifica PUTAUT (CTX), deberá estar autorizado para asumir la identidad de otros usuarios.

**RETRY ( *RecuentoReintentos* | 1 (valor predeterminado) )**

El número de veces, dentro del rango de 1 a 999.999.999, para intentar una acción (en el intervalo especificado en la palabra clave RETRYINT de los datos de control). El número de intentos realizados por el manejador DLQ para implementar una regla concreta es específico de la instancia actual del manejador DLQ; el número total no se conserva al reiniciar. Si el manejador DLQ se reinicia, la cuenta de intentos efectuados para llevar a cabo una regla se restaura a cero.

## Norma de ejemplo

La siguiente es una regla de ejemplo de una tabla de reglas del manejador DLQ:

```
PERSIST(MQPER_PERSISTENT) REASON (MQRC_PUT_INHIBITED) +  
ACTION (RETRY) RETRY (3)
```

Esta regla indica al manejador DLQ que intente tres veces entregar en la cola de destino cualquier mensaje persistente que se haya transferido a la DLQ debido a que MQPUT y MQPUT1 estaban inhibidos.

Todas las palabras clave que pueden utilizarse en una regla están descritas en el resto de este apartado. Tenga en cuenta lo siguiente:

- El valor predeterminado de una palabra clave, si lo hay, está subrayado. Para la mayoría de palabras clave, el valor predeterminado es \* (asterisco), que coincide con cualquier valor.
- La línea vertical (|) separa alternativas, de las que sólo puede especificarse una.
- Todas las palabras clave excepto ACTION son opcionales.

### *Convenios de la tabla de reglas de la cola de mensajes no entregados*

La sintaxis, estructura y contenido de la tabla de reglas del manejador de la cola de mensajes no entregados debe cumplir estos convenios.

La tabla de reglas debe seguir los siguientes convenios:

- Una tabla de reglas debe contener una regla como mínimo.
- Las palabras clave pueden estar en cualquier orden.
- Una palabra clave sólo puede incluirse una vez en cualquier regla.
- Las palabras clave no son sensibles a las mayúsculas y minúsculas.
- Una palabra clave y el valor de su parámetro deben estar separados de las demás palabras clave por un blanco o una coma como mínimo.
- Al principio o al final de una regla y entre palabras clave, signos de puntuación y valores, puede haber cualquier número de espacios en blanco.
- Cada regla debe empezar en una nueva línea.
- En sistemas Windows, la última regla de la tabla debe finalizar con un carácter de retorno de carro/salto de línea. Esto se puede conseguir pulsando la tecla Intro al final de la regla, para que la última línea de la tabla sea una línea en blanco.
- Por motivos de portabilidad, la longitud significativa de una línea no debe ser superior a 72 caracteres.
- Utilice el signo más (+) en una línea como último carácter distinto del blanco para indicar que la regla continúa en el primer carácter distinto del blanco de la línea siguiente. Utilice el signo menos (-) en una línea como último carácter distinto del blanco para indicar que la regla continúa desde el principio de la línea siguiente. Puede haber caracteres de continuación dentro de palabras clave y parámetros.

Por ejemplo:

```
APPLNAME('ABC+  
D')
```

tiene como resultado 'ABCD' y

```
APPLNAME('ABC-  
D')
```

da como resultado 'ABC D'.

- Puede haber líneas de comentario, que empiezan por un asterisco (\*), en cualquier lugar de la tabla de reglas.
- Las líneas en blanco se ignoran.

- Cada entrada de la tabla de reglas del manejador DLQ incluye una o más palabras clave y sus parámetros asociados. Los parámetros deben seguir estas reglas sintácticas:
  - Cada valor de parámetro debe contener, como mínimo, un carácter significativo. Las comillas simples delimitadoras utilizadas en los valores que se encierran entre comillas no se consideran significativas. Por ejemplo, serían válidos estos parámetros:

FORMAT ( ' ABC ' )	3 caracteres significativos
FORMAT ( ABC )	3 caracteres significativos
FORMAT ( ' A ' )	1 carácter significativo
FORMAT ( A )	1 carácter significativo
FORMAT ( ' ' )	1 carácter significativo

Estos parámetros no son válidos porque no contienen caracteres significativos:

```
FORMAT ( ' ' )
FORMAT ( )
FORMAT ( )
FORMAT
```

- Se admiten caracteres comodín. Puede utilizar el signo de interrogación (?) en lugar de cualquier carácter individual, excepto un blanco final; puede utilizar el asterisco (\*) en lugar de cero o más caracteres adyacentes. El asterisco (\*) y el signo de interrogación (?) se interpretan **siempre** como caracteres comodín en los valores de parámetros.
- No pueden incluirse caracteres comodín en los parámetros de las siguientes palabras clave: ACTION, HEADER, RETRY, FWDQ, FWDQM y PUTAUT.
- Los espacios en blanco de cola de los valores de parámetros y de los campos correspondientes en los mensajes de la DLQ no son significativos para realizar comparaciones con comodines. No obstante, los espacios en blanco de cabecera e intercalados en series de caracteres que se encierran entre comillas sí son significativos en las comparaciones con comodines.
- Los parámetros numéricos no pueden incluir el carácter comodín de signo de interrogación (?). Puede utilizar el asterisco (\*) en lugar de un parámetro numérico completo, pero no como parte de un parámetro numérico. Por ejemplo, serían válidos estos parámetro numéricos:

MSGTYPE ( 2 )	Sólo pueden elegirse mensajes de respuesta
MSGTYPE ( * )	Puede elegirse cualquier tipo de mensaje
MSGTYPE ( ' * ' )	Puede elegirse cualquier tipo de mensaje

Sin embargo, MSGTYPE ( ' 2 \* ' ) no es válido, porque incluye un asterisco (\*) como parte de un parámetro numérico.

- Los parámetros numéricos deben estar comprendidos entre 0 y 999.999.999. Si el valor del parámetro está dentro de este rango, se aceptará, incluso si en ese momento no es válido en el campo asociado a la palabra clave. Puede utilizar nombres simbólicos para parámetros numéricos.
- Si un valor de tipo serie es más corto que el campo de MQDLH o MQMD con el que está relacionada la palabra clave, el valor se rellenará con espacios en blanco hasta alcanzar la longitud del campo. Si el valor, excluyendo los asteriscos, es más largo que la longitud del campo, se diagnosticará un error. Por ejemplo, todos estos valores de tipo serie son válidos para un campo de 8 caracteres:

' ABCDEFGH '	8 caracteres
' A * C * E * G * I '	5 caracteres, excluidos los asteriscos
' * A * C * E * G * I * K * M * O * ' *' '	8 caracteres, excluidos los asteriscos



- Escriba series que contengan espacios en blanco, caracteres en minúsculas o caracteres especiales que no sean el punto (.), la barra inclinada (?), el subrayado (\_) y el signo de porcentaje (%) entre comillas simples. Los caracteres en minúsculas que no estén entre comillas simples se convierten a mayúsculas. Si la serie incluye una parte entrecomillada, deberán utilizarse dos comillas simples para indicar el principio y el final de la parte entrecomillada. Cuando se calcula la longitud de la serie, cada aparición de comillas dobles se cuenta como un solo carácter.

### ***Cómo se procesa la tabla de reglas de la cola de mensajes no entregados***

El manejador de la cola de mensajes no entregados busca en la tabla de reglas una regla donde el patrón coincida con un mensaje de la DLQ.

La búsqueda empieza por la primera regla de la tabla y continúa secuencialmente por la tabla. Cuando el manejador DLQ encuentra una regla con un patrón que coincide, intenta realizar la acción que indica la regla. El manejador DLQ aumenta en 1 la cuenta de intentos de la regla cada vez que intenta aplicarla. Si el primer intento falla, el manejador DLQ vuelve a intentarlo hasta que la cuenta de intentos es igual al número especificado en la palabra clave RETRY. Si todos los intentos fallan, el manejador DLQ busca la siguiente regla coincidente de la tabla.

Este proceso se repite para las siguientes reglas coincidentes hasta que se realiza una acción satisfactoriamente. Cuando se han intentado realizar todas las reglas coincidentes el número de veces especificado en su palabra clave RETRY y todos los intentos han fallado, se presupone ACTION (IGNORE). ACTION (IGNORE) también se presupone si no se encuentra ninguna regla coincidente.

#### **Nota:**

1. Sólo se buscan patrones de reglas coincidentes para mensajes de la DLQ que empiecen por MQDLH. Los mensajes que no empiezan por MQDLH se indican periódicamente como erróneos, y permanecen en la DLQ indefinidamente.
2. Todas las palabras clave de patrón pueden adoptar su valor predeterminado, por lo que una regla puede constar únicamente de una acción. No obstante, tenga en cuenta que las reglas que son sólo una acción se aplican a todos los mensajes de la cola que tienen MQDLH y que no se han procesado todavía de acuerdo con otras reglas de la tabla.
3. La tabla de reglas se valida cuando se inicia el manejador DLQ, y los errores se indican en ese momento. Puede efectuar cambios en la tabla de reglas en cualquier momento, pero esos cambios no serán efectivos hasta que se reinicie el manejador DLQ.
4. El manejador DLQ no altera el contenido de los mensajes, de la MQDLH ni del descriptor de mensaje. El manejador DLQ siempre pone mensajes en otras colas con la opción de mensaje MQPMO\_PASS\_ALL\_CONTEXT.
5. Es posible que los errores de sintaxis consecutivos en la tabla de reglas no se reconozcan debido a que la tabla de reglas se ha diseñado para eliminar la generación de errores repetitivos durante la validación.
6. El manejador DLQ abre la DLQ con la opción MQOO\_INPUT\_AS\_Q\_DEF.
7. Pueden ejecutarse simultáneamente varias instancias del manejador DLQ para la misma cola, utilizando la misma tabla de reglas. Pero es más normal que exista una relación unívoca entre una DLQ y un manejador DLQ.

#### **Conceptos relacionados**

Colas de mensajes no entregados

#### **Tareas relacionadas**

Resolución de problemas de mensajes no entregados

#### *Asegurarse de que todos los mensajes DLQ se han procesado*

El manejador de la cola de mensajes no entregados mantiene un registro de todos los mensajes de la DLQ que se han visto pero no eliminado.

Si utiliza el manejador DLQ como filtro para extraer un pequeño subconjunto de mensajes de la DLQ, el manejador DLQ deberá llevar todavía un registro de los mensajes de la DLQ que no haya procesado. El manejador DLQ tampoco puede garantizar que se vayan a ver los nuevos mensajes que lleguen a la DLQ,

incluso si la DLQ está definida como "primero en entrar, primero en salir" (FIFO). Si la cola no está vacía, se realizan nuevas exploraciones periódicas de la DLQ para comprobar todos los mensajes.

Por estas razones, se debe procurar que la DLQ contenga el menor número de mensajes posible; si los mensajes que no se pueden desechar ni reenviar a otras colas (sea cual sea el motivo) pueden acumularse en la cola, la carga de trabajo del manejador DLQ aumenta y la propia DLQ puede llenarse.

Puede tomar medidas específicas que permitan al manejador DLQ vaciar la DLQ. Por ejemplo, intente no utilizar ACTION (IGNORE), que deja los mensajes en la DLQ. (Recuerde que se presupone ACTION (IGNORE) para los mensajes a los que no se aplica explícitamente ninguna otra regla de la tabla). Para estos mensajes que de otro modo se ignorarían, utilice una acción que traslade los mensajes a otra cola. Por ejemplo:

```
ACTION (FWD) FWDQ (IGNORED.DEAD.QUEUE) HEADER (YES)
```

Del mismo modo, haga que la regla final de la tabla procese todos los mensajes en los que no hayan actuado las reglas anteriores de la tabla. Por ejemplo, la regla final de la tabla puede ser parecida a esto:

```
ACTION (FWD) FWDQ (REALLY.DEAD.QUEUE) HEADER (YES)
```

Esto reenvía los mensajes que pasan a la regla final de la tabla a la cola REALLY . DEAD . QUEUE, donde se pueden procesar manualmente. Si no tiene una regla de este tipo, es probable que los mensajes permanezcan en la DLQ indefinidamente.

### ***Ejemplo de tabla de reglas del manejador DLQ***

Un ejemplo de tabla de reglas de la cola de mensajes no entregados para el mandato `runmqdlq`, que contiene una sola entrada de control de datos y varias reglas.

```
*****
*   An example rules table for the runmqdlq command       *
*****
* Control data entry
* -----
* If no queue manager name is supplied as an explicit parameter to
* runmqdlq, use the default queue manager for the machine.
* If no queue name is supplied as an explicit parameter to runmqdlq,
* use the DLQ defined for the local queue manager.
*
inputqm(' ') inputq(' ')

* Rules
* -----
* We include rules with ACTION (RETRY) first to try to
* deliver the message to the intended destination.
* If a message is placed on the DLQ because its destination
* queue is full, attempt to forward the message to its
* destination queue. Make 5 attempts at approximately
* 60-second intervals (the default value for RETRYINT).

REASON(MQRC_Q_FULL) ACTION(RETRY) RETRY(5)

* If a message is placed on the DLQ because of a put inhibited
* condition, attempt to forward the message to its
* destination queue. Make 5 attempts at approximately
* 60-second intervals (the default value for RETRYINT).

REASON(MQRC_PUT_INHIBITED) ACTION(RETRY) RETRY(5)

* The AAAA corporation are always sending messages with incorrect
* addresses. When we find a request from the AAAA corporation,
* we return it to the DLQ (DEADQ) of the reply-to queue manager
* (&REPLYQM).
* The AAAA DLQ handler attempts to redirect the message.

MSGTYPE(MQMT_REQUEST) REPLYQM(AAAA.*) +
ACTION(FWD) FWDQ(DEADQ) FWDQM(&REPLYQM)

* The BBBB corporation never do things by half measures. If
* the queue manager BBBB.1 is unavailable, try to
```

```
* send the message to BBBB.2
```

```
DESTQM(bbbb.1) +  
action(fwd) fwdq(&DESTQ) fwdqm(bbbb.2) header(no)
```

```
* The CCCC corporation considers itself very security  
* conscious, and believes that none of its messages  
* will ever end up on one of our DLQs.  
* Whenever we see a message from a CCCC queue manager on our  
* DLQ, we send it to a special destination in the CCCC organization  
* where the problem is investigated.
```

```
REPLYQM(CCCC.*) +  
ACTION(FWD) FWDQ(ALARM) FWDQM(CCCC.SYSTEM)
```

```
* Messages that are not persistent run the risk of being  
* lost when a queue manager terminates. If an application  
* is sending nonpersistent messages, it should be able  
* to cope with the message being lost, so we can afford to  
* discard the message. PERSIST(MQPER_NOT_PERSISTENT) ACTION(DISCARD)  
* For performance and efficiency reasons, we like to keep  
* the number of messages on the DLQ small.  
* If we receive a message that has not been processed by  
* an earlier rule in the table, we assume that it  
* requires manual intervention to resolve the problem.  
* Some problems are best solved at the node where the  
* problem was detected, and others are best solved where  
* the message originated. We don't have the message origin,  
* but we can use the REPLYQM to identify a node that has  
* some interest in this message.  
* Attempt to put the message onto a manual intervention  
* queue at the appropriate node. If this fails,  
* put the message on the manual intervention queue at  
* this node.
```

```
REPLYQM('?*') +  
ACTION(FWD) FWDQ(DEADQ.MANUAL.INTERVENTION) FWDQM(&REPLYQM)
```

```
ACTION(FWD) FWDQ(DEADQ.MANUAL.INTERVENTION)
```

### Conceptos relacionados

[Colas de mensajes no entregados](#)

### Tareas relacionadas

[Resolución de problemas de mensajes no entregados](#)

### Referencia relacionada

[runmqdlq \(ejecutar manejador de cola de mensajes no entregados\)](#)

## Trabajar con temas administrativos

Utilice los mandatos MQSC para gestionar temas administrativos.

Consulte la sección [Mandatos MQSC](#) para obtener información detallada acerca de estos mandatos.

### Conceptos relacionados

[Objetos de tema administrativo](#)

[“Definición de un tema administrativo” en la página 188](#)

Utilice el mandato MQSC **DEFINE TOPIC** para crear un tema administrativo. Al definir un tema administrativo, puede establecer cada atributo de tema.

[“Visualizar atributos de objeto de tema administrativo” en la página 188](#)

Utilice el mandato MQSC **DISPLAY TOPIC** para visualizar un objeto de tema administrativo.

[“Cambiar atributos de tema administrativo” en la página 189](#)

Puede cambiar los atributos de tema de dos maneras, utilizando el mandato **ALTER TOPIC** o el mandato **DEFINE TOPIC** con el atributo **REPLACE**.

[“Copiar una definición de tema administrativo” en la página 189](#)

Puede copiar una definición de tema utilizando el atributo **LIKE** en el mandato **DEFINE**.

[“Supresión de una definición de tema administrativo” en la página 190](#)

Puede utilizar el mandato MQSC **DELETE TOPIC** para suprimir un tema administrativo.

## Definición de un tema administrativo

Utilice el mandato MQSC **DEFINE TOPIC** para crear un tema administrativo. Al definir un tema administrativo, puede establecer cada atributo de tema.

Los atributos de tema que no se definen explícitamente se heredan del tema administrativo predeterminado, SYSTEM.DEFAULT.TOPIC, creado al realizar la instalación del sistema.

Por ejemplo, el mandato **DEFINE TOPIC** que sigue, define un tema denominado ORANGE . TOPIC con estas características:

- Se resuelve en la serie de tema ORANGE. Para obtener información sobre cómo se pueden utilizar las series de temas, consulte [Combinación de series de temas](#).
- Los atributos establecidos en ASPARENT utilizan el atributo tal como está definido en el tema padre de este tema. Esta acción se repite en el árbol de temas hasta que se encuentra el tema raíz SYSTEM.BASE.TOPIC. Para obtener más información, consulte [Árboles de temas](#).

```
DEFINE TOPIC (ORANGE.TOPIC) +  
TOPICSTR (ORANGE) +  
DEFPRTY(ASPARENT) +  
NPMSGDLV(ASPARENT)
```

### Nota:

- Con la excepción del valor de la serie de tema, todos los valores de atributo que se muestran son los valores predeterminados. Se muestran aquí sólo como ejemplo ilustrativo. Puede omitirlos si está seguro de que los valores predeterminados son los que desea o no han sido modificados. Consulte también [“Visualizar atributos de objeto de tema administrativo”](#) en la página 188.
- Si ya tiene un tema administrativo con el nombre ORANGE.TOPIC en el mismo gestor de colas, este mandato falla. Utilice el atributo REPLACE si desea sobrescribir la definición existente de un tema, pero consulte también [“Cambiar atributos de tema administrativo”](#) en la página 189.

### Referencia relacionada

[DEFINE TOPIC](#)

## Visualizar atributos de objeto de tema administrativo

Utilice el mandato MQSC **DISPLAY TOPIC** para visualizar un objeto de tema administrativo.

Para visualizar todos los temas, utilice:

```
DISPLAY TOPIC (ORANGE.TOPIC)
```

Puede visualizar atributos de forma selectiva especificándolos individualmente con el mandato **DISPLAY TOPIC** . Por ejemplo:

```
DISPLAY TOPIC (ORANGE.TOPIC) +  
TOPICSTR +  
DEFPRTY +  
NPMSGDLV
```

Este mandato muestra los tres atributos especificados de la siguiente manera:

```
AMQ8633: Display topic details.  
TOPIC (ORANGE.TOPIC)                                TYPE (LOCAL)  
TOPICSTR (ORANGE)                                    DEFPRTY (ASPARENT)  
NPMSGDLV (ASPARENT)
```

Para visualizar los valores ASPARENT del tema tal como se utilizan en el tiempo de ejecución, utilice el mandato **DISPLAY TPSTATUS** . Por ejemplo, utilice:

```
DISPLAY TPSTATUS(ORANGE) DEFPRTY NPMSGDLV
```

El mandato muestra los siguientes detalles:

```
AMQ8754: Display topic status details.  
TOPICSTR(ORANGE) DEFPRTY(0)  
NPMSGDLV(ALLAVAIL)
```

Al definir un tema administrativo, éste toma todos los atributos que no especifique explícitamente del tema administrativo predeterminado, que se llama SYSTEM.DEFAULT.TOPIC. Para saber cuáles son estos atributos predeterminados, utilice el siguiente mandato:

```
DISPLAY TOPIC (SYSTEM.DEFAULT.TOPIC)
```

### Referencia relacionada

[DISPLAY TOPIC](#)

[DISPLAY TPSTATUS](#)

## Cambiar atributos de tema administrativo

Puede cambiar los atributos de tema de dos maneras, utilizando el mandato **ALTER TOPIC** o el mandato **DEFINE TOPIC** con el atributo **REPLACE** .

Si, por ejemplo, desea cambiar la prioridad predeterminada de los mensajes entregados a un tema denominado ORANGE.TOPIC, para que sea 5, utilice uno de estos mandatos:

- Utilizando el mandato **ALTER**:

```
ALTER TOPIC(ORANGE.TOPIC) DEFPRTY(5)
```

Este mandato cambia un único atributo, el de la prioridad predeterminada del mensaje entregado a este tema a 5; todos los demás atributos permanecen intactos.

- Utilizando el mandato **DEFINE**:

```
DEFINE TOPIC(ORANGE.TOPIC) DEFPRTY(5) REPLACE
```

Este mandato cambia la prioridad predeterminada de los mensajes entregados a este tema. Todos los demás atributos reciben sus valores predeterminados.

Si modifica la prioridad de los mensajes enviados a este tema, los mensajes existentes no se ven afectados. Los mensajes nuevos, sin embargo, utilizan la prioridad especificada si no la proporciona la aplicación de publicación.

### Referencia relacionada

[ALTER TOPIC](#)

[DISPLAY TOPIC](#)

## Copiar una definición de tema administrativo

Puede copiar una definición de tema utilizando el atributo LIKE en el mandato **DEFINE**.

Por ejemplo:

```
DEFINE TOPIC (MAGENTA.TOPIC) +  
LIKE (ORANGE.TOPIC)
```

Este mandato crea un tema, MAGENTA.TOPIC, con los mismos atributos que el tema original, ORANGE.TOPIC, en lugar de los atributos del tema administrativo predeterminado del sistema. Especifique el nombre del tema que debe copiarse exactamente cómo lo especificó al crear el tema. Si el nombre contiene caracteres en minúscula, especifique el nombre entre comillas simples.

También puede utilizar esta forma del mandato **DEFINE** para copiar una definición de tema, pero realice los cambios en los atributos del original. Por ejemplo:

```
DEFINE TOPIC(BLUE.TOPIC) +  
TOPICSTR(BLUE) +  
LIKE(ORANGE.TOPIC)
```

También puede copiar los atributos del tema BLUE.TOPIC en el tema GREEN.TOPIC y especificar que cuando las aplicaciones no puedan entregarse a la cola de suscriptor correcta no se coloquen en la cola de mensajes no entregados. Por ejemplo:

```
DEFINE TOPIC(GREEN.TOPIC) +  
TOPICSTR(GREEN) +  
LIKE(BLUE.TOPIC) +  
USEDLQ(NO)
```

### Referencia relacionada

[DEFINE TOPIC](#)

## Supresión de una definición de tema administrativo

Puede utilizar el mandato MQSC **DELETE TOPIC** para suprimir un tema administrativo.

Por ejemplo:

```
DELETE TOPIC(ORANGE.TOPIC)
```

Las aplicaciones ya no podrán abrir el tema para su publicación o realizar nuevas suscripciones utilizando el nombre del objeto, ORANGE.TOPIC. Las aplicaciones de publicación que tienen el tema abierto pueden continuar publicando la serie de tema resuelta. Las suscripciones ya realizadas en este tema continúan recibiendo publicaciones después de que se haya suprimido el tema.

Las aplicaciones que no hacen referencia a este objeto de tema, pero que utilizan la serie de tema resuelta representada por este objeto de tema, 'NARANJA' en este ejemplo, siguen funcionando. En este caso heredan las propiedades de un objeto de tema superior en el árbol de temas. Para obtener más información, consulte [Árboles de temas](#).

### Referencia relacionada

[DELETE TOPIC](#)

## Trabajar con suscripciones

Utilice los mandatos MQSC para gestionar suscripciones.

Las suscripciones pueden ser de tres tipos, definidos en el atributo **SUBTYPE**:

### ADMIN

Definida administrativamente por un usuario.

### PROXY

Suscripción creada internamente para direccionar publicaciones entre gestores de colas.

### API

Creada mediante programación, por ejemplo, con la llamada MQI MQSUB.

Consulte la sección [Mandatos MQSC](#) para obtener información detallada acerca de estos mandatos.

### Conceptos relacionados

[“Definir una suscripción administrativa” en la página 191](#)

Utilice el mandato MQSC **DEFINE SUB** para crear una suscripción administrativa. También puede utilizar el valor predeterminado definido en la definición de suscripción local predeterminada. O bien puede modificar las características de suscripción a partir de las de la suscripción local predeterminada SYSTEM.DEFAULT.SUB creada al instalar el sistema.

[“Visualización de atributos de suscripciones” en la página 192](#)

Puede utilizar el mandato **DISPLAY SUB** para visualizar los atributos configurados de cualquier suscripción conocida por el gestor de colas.

[“Cambiar atributos de suscripciones locales” en la página 193](#)

Puede cambiar los atributos de suscripción de dos maneras, utilizando el mandato **ALTER SUB** o el mandato **DEFINE SUB** con el atributo **REPLACE**.

[“Copiar un definición de suscripción local” en la página 193](#)

Puede copiar una definición de suscripción utilizando el atributo **LIKE** en el mandato **DEFINE**.

[“Supresión de una suscripción local” en la página 194](#)

Puede utilizar el mandato MQSC **DELETE SUB** para suprimir una suscripción local.

## Definir una suscripción administrativa

Utilice el mandato MQSC **DEFINE SUB** para crear una suscripción administrativa. También puede utilizar el valor predeterminado definido en la definición de suscripción local predeterminada. O bien puede modificar las características de suscripción a partir de las de la suscripción local predeterminada SYSTEM.DEFAULT.SUB creada al instalar el sistema.

Por ejemplo, el mandato **DEFINE SUB** que sigue define una suscripción denominada ORANGE con estas características:

- Suscripción duradera, lo que significa que persiste tras el reinicio del gestor de colas, con una caducidad ilimitada.
- Recibe las publicaciones creadas en la serie de tema ORANGE, con las prioridades de mensaje tal como las establecen las aplicaciones de publicación.
- Las publicaciones entregadas para esta suscripción se envían a la cola local SUBQ, que debe definirse antes de definir la suscripción.

```
DEFINE SUB (ORANGE) +  
TOPICSTR (ORANGE) +  
DESTCLAS (PROVIDED) +  
DEST (SUBQ) +  
EXPIRY (UNLIMITED) +  
PUBPRTY (AS PUB)
```

### Nota:

- La suscripción y el nombre de la serie de tema no es tienen que coincidir.
- Salvo los valores del destino y la serie de tema, todos los valores de atributo que se muestran son los valores predeterminados. Se muestran aquí sólo como ejemplo ilustrativo. Puede omitirlos si está seguro de que los valores predeterminados son los que desea o no han sido modificados. Consulte también [“Visualización de atributos de suscripciones” en la página 192](#).
- Si ya tiene una suscripción local con el nombre ORANGE en el mismo gestor de colas, este mandato falla. Utilice el atributo **REPLACE** si desea sobrescribir la definición existente de una cola, pero consulte también [“Cambiar atributos de suscripciones locales” en la página 193](#).
- Si la cola SUBQ no existe, este mandato falla.

### Referencia relacionada

[DEFINE SUB](#)

## Visualización de atributos de suscripciones

Puede utilizar el mandato **DISPLAY SUB** para visualizar los atributos configurados de cualquier suscripción conocida por el gestor de colas.

Por ejemplo, utilice:

```
DISPLAY SUB(ORANGE)
```

Puede visualizar de forma selectiva los atributos especificándolos individualmente. Por ejemplo:

```
DISPLAY SUB(ORANGE) +  
SUBID +  
TOPICSTR +  
DURABLE
```

Este mandato muestra los tres atributos especificados de la siguiente manera:

```
AMQ8096: IBM MQ subscription inquired.  
SUBID(414D51204141412020202020202020EE921E4E20002A03)  
SUB(ORANGE) TOPICSTR(ORANGE)  
DURABLE(YES)
```

TOPICSTR es la serie de tema resuelta en la que opera este suscriptor. Cuando se define una suscripción para utilizar un objeto de tema, la serie de tema de dicho objeto se utiliza como prefijo para la serie de tema proporcionada al realizar la suscripción. SUBID es un identificador exclusivo asignado por el gestor de colas cuando se crea una suscripción. Se trata de un atributo útil que se debe visualizar para mostrar porque algunos nombres de suscripciones pueden ser largos o estar en un juego de caracteres diferentes para los que podría resultar poco práctico.

Un método alternativo para visualizar las suscripciones es utilizar el SUBID:

```
DISPLAY SUB +  
SUBID(414D51204141412020202020202020EE921E4E20002A03) +  
TOPICSTR +  
DURABLE
```

Este mandato ofrece la misma salida que antes:

```
AMQ8096: IBM MQ subscription inquired.  
SUBID(414D51204141412020202020202020EE921E4E20002A03)  
SUB(ORANGE) TOPICSTR(ORANGE)  
DURABLE(YES)
```

Las suscripciones de proxy en un gestor de colas no se visualizan de forma predeterminada. Para visualizarlas, especifique el valor PROXY o ALL para **SUBTYPE**.

Puede utilizar el mandato DISPLAY SBSTATUS para ver los atributos de tiempo de ejecución. Por ejemplo, utilice el mandato:

```
DISPLAY SBSTATUS(ORANGE) NUMMSG
```

Se visualiza la siguiente salida:

```
AMQ8099: IBM MQ subscription status inquired.  
SUB(ORANGE)  
SUBID(414D51204141412020202020202020EE921E4E20002A03)  
NUMMSG(0)
```



Al definir una suscripción administrativa, ésta toma los atributos que no se especifican explícitamente de la suscripción predeterminada, que se denomina SYSTEM.DEFAULT.SUB. Para saber cuáles son estos atributos predeterminados, utilice el siguiente mandato:

```
DISPLAY SUB (SYSTEM.DEFAULT.SUB)
```

### Referencia relacionada

[DISPLAY SUB](#)

## Cambiar atributos de suscripciones locales

Puede cambiar los atributos de suscripción de dos maneras, utilizando el mandato **ALTER SUB** o el mandato **DEFINE SUB** con el atributo **REPLACE**.

Si, por ejemplo, desea cambiar a 5 la prioridad de los mensajes entregados a una suscripción denominada ORANGE, utilice uno de estos mandatos:

- Utilizando el mandato **ALTER**:

```
ALTER SUB(ORANGE) PUBPRTY(5)
```

Este mandato cambia un solo atributo, el de la prioridad de los mensajes entregados a esta suscripción a 5; todos los demás atributos permanecen intactos.

- Utilizando el mandato **DEFINE**:

```
DEFINE SUB(ORANGE) PUBPRTY(5) REPLACE
```

Este mandato cambia la prioridad de los mensajes entregados a esta suscripción y todos los demás atributos que tienen a los que se asignan los valores predeterminados.

Si modifica la prioridad de los mensajes enviados a esta suscripción, los mensajes existentes no se verán afectados. Sin embargo, los nuevos mensajes tienen la prioridad especificada.

### Referencia relacionada

[ALTER SUB](#)

[DEFINE SUB](#)

## Copiar un definición de suscripción local

Puede copiar una definición de suscripción utilizando el atributo **LIKE** en el mandato **DEFINE**.

Por ejemplo:

```
DEFINE SUB(BLUE) +  
LIKE(ORANGE)
```

También puede copiar los atributos de la suscripción REAL a la suscripción THIRD.SUB, y especificar el ID de correlación de las publicaciones entregadas como THIRD, en lugar de el ID de correlación de los publicadores. Por ejemplo:

```
DEFINE SUB(THIRD.SUB) +  
LIKE(BLUE) +  
DESTCORL(ORANGE)
```

### Referencia relacionada

[DEFINE SUB](#)

## Supresión de una suscripción local

Puede utilizar el mandato MQSC **DELETE SUB** para suprimir una suscripción local.

```
DELETE SUB(ORANGE)
```

También puede suprimir una suscripción utilizando el SUBID:

```
DELETE SUB SUBID(414D51204141412020202020202020EE921E4E20002A03)
```

### Referencia relacionada

[DELETE SUB](#)

## Comprobación de mensajes en una suscripción

Cuando se define una suscripción, ésta se asocia a una cola. Los mensajes publicados que coinciden con la suscripción se colocan en esta cola.

### Acerca de esta tarea

Tenga en cuenta que los mandatos **runmqsc** siguientes muestran sólo las suscripciones que han recibido mensajes.

Para comprobar los mensajes que actualmente se encuentran en la cola para una suscripción, siga estos pasos:

### Procedimiento

1. Para comprobar si hay mensajes en cola para un tipo de suscripción **DISPLAY SBSTATUS** (*sub\_name*) **NUMMSGS**, consulte [“Visualización de atributos de suscripciones”](#) en la [página 192](#).
2. Si el valor de **NUMMSGS** es mayor que cero, identifique la cola asociada con la suscripción escribiendo **DISPLAY SUB** (*sub\_name*) **DEST**.
3. Utilizando el nombre de la cola devuelta puede ver los mensajes siguiendo la técnica que se describe en [“Examinar colas con el programa de ejemplo”](#) en la [página 157](#).

### Referencia relacionada

[DISPLAY SBSTATUS](#)

## Trabajar con servicios

Los objetos de servicio son una forma de gestionar los procesos adicionales como parte de un gestor de colas. Con los servicios, puede definir programas que se inician y se detienen cuando se inicia o se detiene el gestor de colas. Los servicios de IBM MQ siempre se inician bajo el ID de usuario del usuario que ha iniciado el gestor de colas.

Para definir una nueva definición de servicio de IBM MQ , utilice el mandato MQSC **DEFINE SERVICE**.

Los objetos de servicio pueden ser de los tipos siguientes:

### Servidor

Un servidor es un objeto de servicio que tiene el parámetro **SERVTYPE** especificado como **SERVER**.

Un objeto de servicio de servidor es la definición de un programa que se ejecuta cuando se inicia un gestor de colas especificado. Los objetos de servicio de servidor definen programas que normalmente se ejecutan durante un largo periodo de tiempo. Por ejemplo, un objeto de servicio de servidor se puede utilizar para ejecutar un proceso de supervisor desencadenante, como **runmqtrm**.

Sólo se puede ejecutar una instancia de un objeto de servicio de servidor al mismo tiempo. El estado de los objetos de servicio del servidor que están en ejecución se puede supervisar con el mandato MQSC, **DISPLAY SVSTATUS**.

## Mandato

Un mandato es un objeto de servicio que tiene el parámetro **SERVTYPE** especificado como **COMMAND**. Los objetos de servicio de mandato son similares a los objetos de servicio de servidor, aunque se pueden ejecutar varias instancias de un objeto de servicio de mandato simultáneamente y su estado no se puede supervisar con el mandato MQSC **DISPLAY SVSTATUS**.

Si se ejecuta el mandato MQSC, **STOP SERVICE**, no se realiza ninguna comprobación para determinar si el programa que inició el mandato MQSC, **START SERVICE**, sigue activo antes de detener el programa.

## Referencia relacionada

[DISPLAY SVSTATUS](#)

[START SERVICE](#)

[STOP SERVICE](#)

## Definición de un objeto de servicio

Puede definir un objeto de servicio con el mandato MQSC **DEFINE SERVICE**.

Los atributos que necesita definir son los siguientes:

### SERVTYPE

Define el tipo del objeto de servicio. Los valores posibles son los siguientes:

#### Servidor

Objeto de servicio del servidor.

Sólo se puede ejecutar una instancia de un objeto de servicio de servidor al mismo tiempo. El estado de los objetos de servicio de servidor se puede supervisar utilizando el mandato MQSC, **DISPLAY SVSTATUS**.

#### Mandato

Objeto de servicio del mandato.

Se pueden ejecutar varias instancias de un objeto de servicio de mandato al mismo tiempo. El estado de un objeto de servicio de mandato no se pueden supervisar.

### STARTCMD

El programa que se ejecuta para iniciar el servicio. Debe especificarse una vía de acceso completa al programa.

### STARTARG

Argumentos que se han pasado al programa de inicio.

### STDERR

Especifica la vía de acceso a un archivo al que se debería redirigir la salida de error estándar (stderr) del programa de servicio.

### STDOUT

Especifica la vía de acceso a un archivo al que se debería redirigir la salida estándar (stdout) del programa de servicio.

### STOPCMD

El programa que se ejecuta para detener el servicio. Debe especificarse una vía de acceso completa al programa.

### STOPARG

Argumentos que se han pasado al programa de detención.

### CONTROL

Especifica cómo se debe iniciar y detener el servicio.

### MANUAL

El servicio no se debe iniciar ni detener de forma automática. Se controla mediante los mandatos **START SERVICE** y **STOP SERVICE**. Éste es el valor predeterminado.

## QMGR

El servicio que se define se debe iniciar y detener al mismo tiempo que se inicia y se detiene el gestor de colas.

## STARTONLY

El servicio debe iniciarse al mismo tiempo que se inicia el gestor de colas, pero no tiene que detenerse cuando se detiene el gestor de colas.

### Conceptos relacionados

“Gestión de servicios” en la página 196

Mediante la utilización del parámetro **CONTROL**, una instancia de un objeto de servicio se puede iniciar o detener automáticamente con el gestor de colas, o se puede iniciar y detener mediante los mandatos MQS MQSC **START SERVICE** y **STOP SERVICE**.

### Referencia relacionada

[DEFINE SERVICE](#)

[DISPLAY SVSTATUS](#)

[START SERVICE](#)

[STOP SERVICE](#)

## Gestión de servicios

Mediante la utilización del parámetro **CONTROL**, una instancia de un objeto de servicio se puede iniciar o detener automáticamente con el gestor de colas, o se puede iniciar y detener mediante los mandatos MQS MQSC **START SERVICE** y **STOP SERVICE**.

Cuando se inicia una instancia de un objeto de servicio, se graba un mensaje en las anotaciones de error del gestor de colas, que contiene el nombre del objeto de servicio y el ID de proceso del proceso iniciado. A continuación se muestra una entrada de anotaciones de ejemplo para un objeto de servicio de servidor que se está iniciando:

```
02/15/2005 11:54:24 AM - Process(10363.1) User(mqm) Program(amqzmgr0)
Host(HOST_1) Installation(Installation1)
VRMF(7.1.0.0) QMgr(A.B.C)
AMQ5028: The Server 'S1' has started. ProcessId(13031).
```

```
EXPLANATION:
The Server process has started.
ACTION:
None.
```

A continuación se muestra una entrada de anotaciones de ejemplo para un objeto de servicio de mandato que se está iniciando:

```
02/15/2005 11:53:55 AM - Process(10363.1) User(mqm) Program(amqzmgr0)
Host(HOST_1) Installation(Installation1)
VRMF(7.1.0.0) QMgr(A.B.C)
AMQ5030: The Command 'C1' has started. ProcessId(13030).
```

```
EXPLANATION:
The Command has started.
ACTION:
None.
```

Cuando se detiene una instancia de un servicio de servidor, se graba un mensaje en las anotaciones de error del gestor de colas, que contiene el nombre del servicio y el ID de proceso del proceso que finaliza. A continuación se muestra una entrada de anotaciones de ejemplo para un objeto de servicio de servidor que se está deteniendo:

```
02/15/2005 11:54:54 AM - Process(10363.1) User(mqm) Program(amqzmgr0)
Host(HOST_1) Installation(Installation1)
VRMF(7.1.0.0) QMgr(A.B.C)
AMQ5029: The Server 'S1' has ended. ProcessId(13031).
```

```
EXPLANATION:
```

```
The Server process has ended.  
ACTION:  
None.
```

## Referencia relacionada

[“Variables de entorno adicionales” en la página 197](#)

Cuando se inicia un servicio, el entorno en el que se inicia el proceso de servicio se hereda del entorno del gestor de colas. Es posible definir variables de entorno adicionales que se deben establecer en el entorno del proceso de servicio añadiendo las variables que desea definir en uno de los archivos de alteración temporal del entorno `service.env`.

[STOP SERVICE](#)

[START SERVICE](#)

## Variables de entorno adicionales




Cuando se inicia un servicio, el entorno en el que se inicia el proceso de servicio se hereda del entorno del gestor de colas. Es posible definir variables de entorno adicionales que se deben establecer en el entorno del proceso de servicio añadiendo las variables que desea definir en uno de los archivos de alteración temporal del entorno `service.env`.

## Archivos a los que puede añadir variables de entorno

Puede añadir las variables de entorno a dos archivos posibles:




### El archivo de ámbito de máquina `service.env`

Este archivo se encuentra en:

-   `/var/mqm` en sistemas UNIX and Linux.
-  El directorio de datos seleccionado durante la instalación en los sistemas Windows.

### El archivo `service.env` del ámbito del gestor de colas

Este archivo se encuentra en el directorio de datos del gestor de colas. Por ejemplo, la ubicación del archivo de alteración temporal de entorno para un gestor de colas denominado QMNAME es:

-   En sistemas UNIX and Linux, `/var/mqm/qmgrs/QMNAME/service.env`
-  En sistemas Windows,  
`C:\ProgramData\IBM\MQ\qmgrs\QMNAME\service.env`

Los dos archivos se procesan, si están disponibles, y las definiciones del archivo del ámbito de gestor de colas tienen preferencia sobre las del archivo del ámbito de máquina.

## Variables de entorno que se pueden especificar en `service.env`.

Cualquier variable de entorno se puede especificar en `service.env`. Por ejemplo, si el servicio IBM MQ ejecuta varios mandatos, puede ser útil establecer la variable de usuario `PATH` en el archivo `service.env`. Los valores que establece la variable no pueden ser variables de entorno; por ejemplo, `CLASSPATH= %CLASSPATH%` es incorrecto. Del mismo modo, en Linux `PATH= $PATH : /opt/mqm/bin` daría resultados inesperados.

`CLASSPATH` debe estar en mayúsculas y la sentencia de vía de acceso de clase sólo puede contener literales. Algunos servicios (telemetría por ejemplo) establecen su propia vía de acceso de clase. La `CLASSPATH` definida en `service.env` se añade a la misma.

El formato de las variables definidas en el archivo, `service.env` es una lista de pares de variables de nombre y valor. Cada variable debe definirse en una línea nueva, y cada variable se toma tal como se define explícitamente, incluidos los espacios en blanco.

## Ejemplo de service.env

```
#####  
##  
## <N_OCO_COPYRIGHT> ##  
## Licensed Materials - Property of IBM ##  
## ##  
## 63H9336 ##  
## (C) Copyright IBM Corporation 2005, 2022. ##  
## ##  
## <NOC_COPYRIGHT> ##  
## ##  
#####  
## ***** ##  
## Module Name: service.env ##  
## Type : IBM MQ service environment file ##  
## Function : Define additional environment variables to be set ##  
## for SERVICE programs. ##  
## Usage : <VARIABLE>=<VALUE> ##  
## ##  
## ***** ##  
MYLOC=/opt/myloc/bin  
MYTMP=/tmp  
TRACEDIR=/tmp/trace  
MYINITQ=ACCOUNTS.INITIATION.QUEUE
```

### Referencia relacionada

“Inserciones reemplazables en definición de servicio” en la página 198

En la definición de un objeto de servicio, es posible sustituir las señales. Los tokens que se sustituyan se reemplazan automáticamente por su texto ampliado cuando se ejecute el programa de servicio. Los tokens de sustitución se pueden obtener de la lista siguiente de señales comunes o de las variables definidas en el archivo service.env.

### Inserciones reemplazables en definición de servicio

En la definición de un objeto de servicio, es posible sustituir las señales. Los tokens que se sustituyan se reemplazan automáticamente por su texto ampliado cuando se ejecute el programa de servicio. Los tokens de sustitución se pueden obtener de la lista siguiente de señales comunes o de las variables definidas en el archivo service.env.




A continuación se muestran tokens comunes que se pueden utilizar para sustituir tokens en la definición de un objeto de servicio:

#### MQ\_INSTALL\_PATH

La ubicación donde se ha instalado IBM MQ.

#### MQ\_DATA\_PATH

La ubicación del directorio de datos de IBM MQ:

-   En sistemas UNIX and Linux, la ubicación del directorio de datos de IBM MQ es /var/mqm/
-  En sistemas Windows, la ubicación del directorio de datos de IBM MQ es el directorio de datos seleccionado durante la instalación de IBM MQ

#### QMNAME

El nombre del gestor de colas actual.

#### MQ\_SERVICE\_NAME

Nombre del servicio.

#### MQ\_SERVER\_PID

Esta señal solo la pueden utilizar los argumentos **STOPARG** y **STOPCMD**.

Para objetos de servicio de servidor, esta señal se reemplaza con el ID de proceso del proceso iniciado por los argumentos **STARTCMD** y **STARTARG**. De lo contrario, esta señal se reemplaza por 0.

#### MQ\_Q\_MGR\_DATA\_PATH

La ubicación del directorio de datos del gestor de colas.

## MQ\_Q\_MGR\_DATA\_NAME

El nombre transformado del gestor de colas. Para obtener más información sobre la transformación de nombres, consulte [Comprender los nombres de archivo IBM MQ](#).

Si desea utilizar inserciones reemplazables, inserte la señal entre caracteres + en cualquiera de las series de **STARTCMD**, **STARTARG**, **STOPCMD**, **STOPARG**, **STDOUT** o **STDERR**. Si desea obtener ejemplos, consulte el apartado [“Ejemplos de utilización de objetos de servicio”](#) en la página 199.

## Ejemplos de utilización de objetos de servicio

Los servicios que aparecen en esta sección se escriben con caracteres separadores de vía de acceso de estilo UNIX, excepto cuando se indique lo contrario.

### Utilización de un objeto de servicio de servidor

En este ejemplo se muestra cómo definir, utilizar y modificar un objeto de servicio de servidor para iniciar un supervisor desencadenante.

1. Se define un objeto de servicio de servidor mediante el mandato MQSC **DEFINE SERVICE**:

```
DEFINE SERVICE(S1) +
CONTROL(QMGR) +
SERVTYPE(SERVER) +
STARTCMD('+MQ_INSTALL_PATH+bin/runmqtrm') +
STARTARG('-m +QMNAME+ -q ACCOUNTS.INITIATION.QUEUE') +
STOPCMD('+MQ_INSTALL_PATH+bin/amqsstop') +
STOPARG('-m +QMNAME+ -p +MQ_SERVER_PID+')
```

Donde:

+MQ\_INSTALL\_PATH+ es una señal que representa el directorio de instalación.

+QMNAME+ es una señal que representa el nombre del gestor de colas.

ACCOUNTS.INITIATION.QUEUE es la cola de inicio.

amqsstop es un programa de ejemplo que se proporciona con IBM MQ y que solicita al gestor de colas que interrumpa todas las conexiones del ID de proceso. amqsstop genera mandatos PCF, por lo que el servidor de mandatos debe estar en ejecución.

+MQ\_SERVER\_PID+ es un token que representa el ID de proceso que se pasa al programa de parada.

Consulte [“Inserciones reemplazables en definición de servicio”](#) en la página 198 para ver una lista de las señales habituales.

2. La próxima vez que se inicie el gestor de colas, se ejecutará una instancia del objeto de servicio del servidor. Sin embargo, iniciaremos una instancia del objeto de servicio de servidor inmediatamente con el mandato MQSC **START SERVICE** :

```
START SERVICE(S1)
```

3. Se visualiza el estado del proceso de servicio del servidor, utilizando el mandato MQSC de **DISPLAY SVSTATUS** :

```
DISPLAY SVSTATUS(S1)
```

4. Este ejemplo ahora muestra cómo modificar el objeto de servicio de servidor y cómo conseguir que las actualizaciones se identifiquen y reciban reiniciando manualmente el proceso de servicio de servidor. El objeto de servicio de servidor se modifica para que la cola de inicio se especifique como JUPITER.INITIATION.QUEUE. Se utiliza el mandato MQSC **ALTER SERVICE**:

```
ALTER SERVICE(S1) +
STARTARG('-m +QMNAME+ -q JUPITER.INITIATION.QUEUE')
```

**Nota:** Un servicio en ejecución no identificará ninguna actualización en la definición de servicio hasta que se reinicie.

5. El proceso de servicio de servidor se reinicia para que se aplique la modificación mediante los mandatos MQSC **STOP SERVICE** y **START SERVICE**:

```
STOP SERVICE(S1)
```

Seguidos de:

```
START SERVICE(S1)
```

El proceso de servicio de servidor se reinicia e identifica las modificaciones realizadas en el paso “4” en la página 199.

**Nota:** El mandato MQSC, **STOP SERVICE**, solo se puede utilizar si se especifica un argumento **STOPCMD** en la definición de servicio.

### Referencia relacionada

[ALTER SERVICE](#)

[DEFINE SERVICE](#)

[DISPLAY SVSTATUS](#)

[START SERVICE](#)

[STOP SERVICE](#)

### **Utilización de un objeto de servicio de mandato**

En este ejemplo se muestra cómo definir un objeto de servicio de mandato para iniciar un programa que graba entradas en el registro del sistema del sistema operativo cuando se inicia o se detiene un gestor de colas.

1. El objeto de servicio de mandatos se define mediante el mandato MQSC **DEFINE SERVICE**:

```
DEFINE SERVICE(S2) +  
CONTROL(QMGR) +  
SERVTYPE(COMMAND) +  
STARTCMD('/usr/bin/logger') +  
STARTARG('Queue manager +QMNAME+ starting') +  
STOPCMD('/usr/bin/logger') +  
STOPARG('Queue manager +QMNAME+ stopping')
```

Donde:

logger es el mandato que proporciona el sistema UNIX and Linux para grabar en registro del sistema.

+QMNAME+ es una señal que representa el nombre del gestor de colas.

### Referencia relacionada

[DEFINE SERVICE](#)

### **Utilización de un objeto de servicio de mandato cuando un gestor de colas sólo finaliza**

En este ejemplo se muestra cómo definir un objeto de servicio de mandato para iniciar un programa que graba entradas en el registro del sistema del sistema operativo cuando un gestor de colas sólo se detiene.

1. El objeto de servicio de mandatos se define mediante el mandato MQSC **DEFINE SERVICE**:

```
DEFINE SERVICE(S3) +  
CONTROL(QMGR) +  
SERVTYPE(COMMAND) +  
STOPCMD('/usr/bin/logger') +  
STOPARG('Queue manager +QMNAME+ stopping')
```

Donde:



logger es un programa de ejemplo que se proporciona con IBM MQ que puede grabar entradas en el registro del sistema del sistema operativo.

+QMNAME+ es una señal que representa el nombre del gestor de colas.

### Referencia relacionada

[DEFINE SERVICE](#)

### **Más información sobre el paso de argumentos**

En este ejemplo se muestra cómo definir un objeto de servicio de servidor para iniciar un programa llamado `runserv` cuando se inicie un gestor de colas.

Este ejemplo se ha escrito con caracteres separadores de vías de acceso de estilo Windows.

Uno de los argumentos que se va a pasar al programa que se está iniciando es una serie que contiene un espacio. Este argumento debe pasarse como una sola serie de caracteres; Para ello, se utilizan comillas dobles, tal como se muestra en el mandato siguiente, para definir el objeto de servicio de mandato:

1. El objeto de servicio de servidor se define con el mandato MQSC **DEFINE SERVICE**:

```
DEFINE SERVICE(S1) SERVTYPE(SERVER) CONTROL(QMGR) +
STARTCMD('C:\Program Files\Tools\runserv.exe') +
STARTARG('-m +QMNAME+ -d "C:\Program Files\Tools\'') +
STDOUT('C:\Program Files\Tools\+MQ_SERVICE_NAME+.out')

DEFINE SERVICE(S4) +
CONTROL(QMGR) +
SERVTYPE(SERVER) +
STARTCMD('C:\Program Files\Tools\runserv.exe') +
STARTARG('-m +QMNAME+ -d "C:\Program Files\Tools\'') +
STDOUT('C:\Program Files\Tools\+MQ_SERVICE_NAME+.out')
```

Donde:

+QMNAME+ es una señal que representa el nombre del gestor de colas.

"C:\Program Files\Tools\'" es una serie que contiene un espacio, que se pasará como una única serie.

### Referencia relacionada

[DEFINE SERVICE](#)

### **Inicio automático de un servicio**

En este ejemplo se muestra cómo definir un objeto de servicio del servidor para que inicie automáticamente el supervisor de activaciones cuando se inicie el gestor de colas.

1. El objeto de servicio de servidor se define con el mandato MQSC **DEFINE SERVICE**:

```
DEFINE SERVICE(TRIG_MON_START) +
CONTROL(QMGR) +
SERVTYPE(SERVER) +
STARTCMD('runmqtrm') +
STARTARG('-m +QMNAME+ -q +IQNAME+')
```

Donde:

+QMNAME+ es una señal que representa el nombre del gestor de colas.

+IQNAME+ es una variable de entorno que define el usuario en uno de los archivos `service.env` y que representa el nombre de la cola de inicio.

### Referencia relacionada

[DEFINE SERVICE](#)

## Gestión de objetos para desencadenamiento

IBM MQ le permite iniciar una aplicación automáticamente cuando se cumplen determinadas condiciones en una cola. Por ejemplo, es posible que desee iniciar una aplicación cuando el número de mensaje de una cola alcanza un número especificado. Este recurso se denomina *desencadenamiento*. Deberá definir los objetos que dan soporte al desencadenamiento.

El desencadenamiento se describe detalladamente en [Inicio de aplicaciones IBM MQ utilizando desencadenantes](#).

### Definición de una cola de aplicación para el desencadenamiento

Una cola de aplicación es una cola local que las aplicaciones utilizan para el envío de mensajes, mediante la MQI. El desencadenamiento requiere definir varios atributos de cola en la cola de aplicación.

La activación en sí se habilita mediante el atributo **Trigger** (TRIGGER en los mandatos MQSC). En este ejemplo, se va a generar un suceso desencadenante cuando haya 100 mensajes con una prioridad 5 o superior en la cola local MOTOR.INSURANCE.QUEUE, como se indica a continuación:

```
DEFINE QLOCAL (MOTOR.INSURANCE.QUEUE) +  
PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS) +  
MAXMSGL (2000) +  
DEFPSIST (YES) +  
INITQ (MOTOR.INS.INIT.QUEUE) +  
TRIGGER +  
TRIGTYPE (DEPTH) +  
TRIGDPTH (100)+  
TRIGMPRI (5)
```

donde:

#### **QLOCAL (MOTOR.INSURANCE.QUEUE)**

Es el nombre de la cola de aplicación que se define.

#### **PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS)**

Es el nombre de la definición de proceso que describe la aplicación que se va a iniciar mediante un programa supervisor desencadenante.

#### **MAXMSGL (2000)**

Especifica la longitud máxima de los mensajes de la cola.

#### **DEFPSIST (YES)**

Especifica que los mensajes de esta cola son persistentes de forma predeterminada.

#### **INITQ (MOTOR.INS.INIT.QUEUE)**

Es el nombre de la cola de inicio a la que el gestor de colas va a transferir el mensaje desencadenante.

#### **TRIGGER**

Es el valor del atributo de desencadenamiento.

#### **TRIGTYPE (DEPTH)**

Especifica que se genera un suceso desencadenante cuando el número de mensajes de la prioridad obligatoria (TRIGMPRI) alcanza el número especificado en TRIGDPTH.

#### **TRIGDPTH (100)**

Especifica el número de mensajes necesarios para generar un suceso desencadenante.

#### **TRIGMPRI (5)**

Es la prioridad de los mensajes que el gestor de colas va a incluir en el recuento para decidir si va a generar un suceso desencadenante. Sólo se incluyen en el recuento los mensajes con prioridad 5 o superior.

### Definición de una cola de inicio

Cuando se produce un suceso desencadenante, el gestor de colas coloca un mensaje desencadenante en la cola de inicio especificada en la definición de la cola de aplicación. Las colas de inicio no tienen valores especiales, pero puede utilizar como guía la siguiente definición de la cola local MOTOR.INS.INIT.QUEUE:

```
DEFINE QLOCAL(MOTOR.INS.INIT.QUEUE) +
GET (ENABLED) +
NOSHARE +
NOTRIGGER +
MAXMSGL (2000) +
MAXDEPTH (1000)
```

## Definición de un proceso

Utilice el mandato DEFINE PROCESS para crear una definición de proceso. Una definición de proceso define la aplicación que se va a utilizar para procesar los mensajes de la cola de aplicación. La definición de cola de aplicación nombra el proceso que se va a utilizar y, de este modo, asocia la cola de aplicación con la aplicación que se va a utilizar para procesar sus mensajes. Esto se hace mediante el atributo PROCESS en la cola de aplicación MOTOR.INSURANCE.QUEUE. El siguiente mandato MQSC define el proceso necesario, MOTOR.INSURANCE.QUOTE.PROCESS, identificado en este ejemplo:

```
DEFINE PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS) +
DESCR ('Insurance request message processing') +
APPLTYPE (UNIX) +
APPLICID ('/u/admin/test/IRMP01') +
USERDATA ('open, close, 235')
```

Donde:

### **MOTOR.INSURANCE.QUOTE.PROCESS**

Es el nombre de la definición de proceso.

### **DESCR ('Insurance request message processing')**

Describe el programa de aplicación relacionado con esta definición. Este texto Aparece cuando utiliza el mandato DISPLAY PROCESS y sirve para ayudarle a identificar lo que hace el proceso. Si utiliza espacios en la serie, debe colocarla entre comillas simples.

### **APPLTYPE (UNIX)**

Es el tipo de aplicación que debe iniciarse.

### **APPLICID ('/u/admin/test/IRMP01')**

Es el nombre del archivo ejecutable de la aplicación, especificado como nombre de archivo totalmente calificado. En sistemas Windows, un valor típico de APPLICID sería c:\appl\test\irmp01.exe.

### **USERDATA ('open, close, 235')**

Son datos definidos por el usuario, que la aplicación puede utilizar.

## Visualización de los atributos de una definición de proceso

Utilice el mandato DISPLAY PROCESS para examinar los resultados de la definición. Por ejemplo:

```
DISPLAY PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS)

24 : DISPLAY PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS) ALL
AMQ8407: Display Process details.
DESCR ('Insurance request message processing')
APPLICID ('/u/admin/test/IRMP01')
USERDATA (open, close, 235)
PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS)
APPLTYPE (UNIX)
```

También puede utilizar el mandato MQSC ALTER PROCESS para modificar una definición de proceso existente, y el mandato DELETE PROCESS para suprimir una definición de proceso.

## Utilización del programa de utilidad **dmpmqmsg** entre dos sistemas

El programa de utilidad **dmpmqmsg** (anteriormente *qload*) se ha incorporado en el producto desde IBM MQ 8.0. Anteriormente el programa de utilidad estaba disponible como el SupportPac MO03.

### Visión general

El programa de utilidad **dmpmqmsg** le permite copiar o mover el contenido de una cola, o sus mensajes, a un archivo. Este archivo se puede guardar, según sea necesario, y utilizar posteriormente para volver a cargar los mensajes en la cola.

**Importante:** El archivo tiene un formato específico que comprende el programa de utilidad. Sin embargo, el archivo también pueden leerlo los usuarios, de modo que puede actualizarlo en un editor antes de volver a cargarlo. Si edita el archivo no debe modificar su formato.

Los usos posibles son:

- Guardar en un archivo los mensajes que están en una cola. Posiblemente para fines de archivado y para volver a cargarlo en una cola posteriormente.
- Volver a cargar una cola con los mensajes previamente guardados en un archivo.
- Eliminar los mensajes antiguos de una cola.
- 'Reproducir' los mensajes de prueba desde una ubicación almacenada, incluso manteniendo el tiempo correcto entre los mensajes, si es necesario.



**Atención:** El SupportPac MO03 utilizaba el parámetro **-l** para especificar enlace local o de cliente. El parámetro **-l** se ha sustituido por el parámetro **-c**.

**-P** se utiliza ahora para la información de página de códigos en lugar del parámetro **-c**.

Consulte [dmpmqmsg](#) para obtener más información sobre el mandato y los parámetros disponibles.

### Ejemplo de utilización del programa de utilidad **dmpmqmsg** en Linux, mediante una máquina de Windows

Tiene un gestor de colas en una máquina de Linux que tiene mensajes en una cola (*Q1*) que desea mover a otra cola (*Q2*) del mismo gestor de colas. Desea iniciar el programa de utilidad **dmpmqmsg** desde una máquina Windows.

La cola (*Q1*) tiene cuatro mensajes que se han añadido mediante la aplicación **amqspu**t (gestor de colas local) de ejemplo o la aplicación **amqspu**tc (gestor de colas remoto).

En la máquina de Linux puede ver:

```
display ql(Q1) CURDEPTH
  2 : display ql(Q1) CURDEPTH
AMQ8409: Display Queue details.
  QUEUE(Q1)
  TYPE(LOCAL)
  CURDEPTH(4)
```

Establezca la variable de entorno MQSERVER para que apunte al gestor de colas en Linux. Por ejemplo:

```
set MQSERVER=SYSTEM.DEF.SVRCONN/TCP/veracruz.x.com(1414)
```

donde *veracruz* es el nombre de la máquina.

Ejecute el programa de utilidad **dmpmqmsg** para leer en la cola, *Q1*, y almacenar la salida en `c:\temp\mqqload.txt`.

Conéctese como un cliente remoto al gestor de colas, *QM\_VER*, que se ejecuta en el host y el puerto Linux establecidos por MQSERVER. Puede establecer la conexión como cliente remoto utilizando el atributo: -c.

```
dmpmqmsg -m QM_VER -i Q1 -f c:\temp\mqqlload.txt -c
Read      - Files:    0  Messages:    4  Bytes:          22
Written - Files:    1  Messages:    4  Bytes:          22
```

El archivo de salida `c:\temp\mqqlload.txt` contiene texto con un formato que el programa de utilidad **dmpmqmsg** entiende.

En la máquina de Windows, emita el mandato **dmpmqmsg** (utilizando la opción -o en lugar de la opción -i) para cargar la cola (Q2) en la máquina de Linux desde un archivo en la máquina de Windows:

```
dmpmqmsg -m QM_VER -o Q2 -f c:\temp\mqqlload.txt -c
Read      - Files:    1  Messages:    4  Bytes:          22
Written - Files:    0  Messages:    4  Bytes:          22
```

En la máquina de Linux, observe que ahora hay cuatro mensajes en la cola que se han restaurado a partir del archivo.

```
display ql(Q2) CURDEPTH
      6 : display ql(Q2) CURDEPTH
AMQ8409: Display Queue details.
      QUEUE(Q2)
      TYPE(LOCAL)
      CURDEPTH(4)
```

En la máquina de Linux,

Suprima los mensajes de la cola original.

```
clear qllocal(Q1)
      4 : clear qllocal(Q1)
AMQ8022: IBM MQ queue cleared.
```

Confirme que no haya más mensajes en la cola original:

```
display ql(Q1) CURDEPTH
      5 : display ql(Q1) CURDEPTH
AMQ8409: Display Queue details.
      QUEUE(Q1)
      TYPE(LOCAL)
      CURDEPTH(0)
```

Consulte [dmpmqmsg](#) para ver una descripción del mandato y sus parámetros.

### Conceptos relacionados

“Ejemplos de utilización del programa de utilidad **dmpmqmsg**” en la página 205

Formas sencillas de utilizar el programa de utilidad **dmpmqmsg** (antes **qload**). Esta utilidad está incorporada en el producto desde IBM MQ 8.0.

### Ejemplos de utilización del programa de utilidad **dmpmqmsg**

Formas sencillas de utilizar el programa de utilidad **dmpmqmsg** (antes **qload**). Esta utilidad está incorporada en el producto desde IBM MQ 8.0.

Anteriormente, el programa de utilidad **qload** estaba disponible como SupportPac MO03.

### Descargar una cola en un archivo

Utilice las siguientes opciones de la línea de mandatos para guardar en un archivo los mensajes que están en una cola:

```
dmpmqmsg -m QM1 -i Q1 -f c:\myfile
```

Este mandato realiza una copia de los mensajes de la cola y los guarda en el archivo especificado.

## Descargar una cola en una serie de archivos

Puede descargar una cola en una serie de archivos utilizando un carácter `insert` en el nombre de archivo. En esta modalidad cada mensaje se graba en un archivo nuevo:

```
dmpmqmsg -m QM1 -i Q1 -f c:\myfile%n
```

Este mandato descarga la cola en los archivos, `myfile1`, `myfile2`, `myfile3`, etc.

## Cargar una cola desde un archivo

Para volver a cargar una cola con los mensajes que ha guardado en [“Descargar una cola en un archivo”](#) en la [página 205](#), utilice las opciones siguientes en la línea de mandatos:

```
dmpmqmsg -m QM1 -o Q1 -f c:\myfile%n
```

Este mandato descarga la cola en los archivos, `myfile1`, `myfile2`, `myfile3`, etc.

## Cargar una cola desde una serie de archivos

Puede cargar una cola desde una serie de archivos utilizando un carácter `insert` en el nombre de archivo. En esta modalidad cada mensaje se graba en un archivo nuevo:

```
dmpmqmsg -m QM1 -o Q1 -f c:\myfile%n
```

Este mandato carga la cola en los archivos, `myfile1`, `myfile2`, `myfile3`, etc.

## Copiar los mensajes de una cola a otra cola

Sustituya el parámetro del archivo en [“Descargar una cola en un archivo”](#) en la [página 205](#), con otro nombre de cola y utilice las opciones siguientes:

```
dmpmqmsg -m QM1 -i Q1 -o Q2
```

Este mandato permite copiar los mensajes de una cola en otra cola.

## Copiar los primeros 100 de una cola a otra cola

Utilice el mandato del ejemplo anterior y añada la opción `-r#100` :

```
dmpmqmsg -m QM1 -i Q1 -o Q2 -r#100
```

## Mover los mensajes de una cola a otra cola

Una variación de [“Cargar una cola desde un archivo”](#) en la [página 206](#). Observe la diferencia entre utilizar `-i` (minúsculas) que solo examina una cola y `-I` (mayúsculas) que obtiene de forma destructiva de una cola:

```
dmpmqmsg -m QM1 -I Q1 -o Q2
```

## Mover de una cola a otra cola los mensajes con una antigüedad de más de un día

Este ejemplo muestra cómo se utiliza la selección de antigüedad. Se pueden seleccionar mensajes con una antigüedad mayor que o menor que, o en un rango de antigüedad.

```
dmpmqmsg -m QM1 -I Q1 -o Q2 -T1440
```

## Mostrar la antigüedad de los mensajes que están actualmente en la cola

Utilice las opciones siguientes en la línea de mandatos:

```
dmpmqmsg -m QM1 -i Q1 -f stdout -dT
```

## Trabajar con el archivo de mensajes

Después de descargar el mensaje desde una cola, como en el paso [“Descargar una cola en un archivo”](#) en la [página 205](#), es posible que desee editar el archivo.

También es posible que desee cambiar el formato del archivo para utilizar una de las opciones de visualización que no ha especificado en el momento de descargar la cola.

Puede utilizar la utilidad **dmpmqmsg** para volver a procesar el archivo en el formato necesario incluso después de haber tenido lugar la descarga de la cola. Utilice las opciones siguientes en la línea de mandatos.

```
dmpmqmsg -f c:\oldfile -f c:\newfile -dA
```

Consulte [dmpmqmsg](#) para ver una descripción del mandato y sus parámetros.

## Trabajar con objetos de IBM MQ remotos

Puede administrar objetos IBM MQ en gestor de colas remotos utilizando mandatos MQSC, mandatos PCF o la administrativa REST API. Antes de poder utilizar cualquiera de estos métodos, debe definir colas de transmisión y canales entre el gestor de colas local y el gestor de colas remoto, para que los mandatos se puedan enviar al gestor de colas remoto y las respuestas recibidas por el gestor de colas local. De forma alternativa, puede configurar un clúster de gestor de colas y, después, utilizar los mismos métodos de administración remota.

### Acerca de esta tarea

Para preparar los gestores de colas para la administración remota, debe configurar los objetos siguientes en el gestor de colas local:

- Un escucha
- Una cola de transmisión que tiene el nombre del gestor de colas remoto.
- Un canal emisor que tiene los detalles de la conexión para el gestor de colas remoto.
- Un canal receptor que tiene el mismo nombre que el canal emisor en el gestor de colas remoto.

También debe configurar los objetos siguientes en el gestor de colas remoto:

- Un escucha
- Una cola de transmisión que tiene el nombre del gestor de colas local.
- Un canal emisor que tiene los detalles de conexión para el gestor de colas local.
- Un canal receptor que tiene el mismo nombre que el canal emisor en el gestor de colas local.

Si desea más información sobre cómo configurar estos objetos, consulte [“Configuración de gestores de colas para la administración remota”](#) en la [página 208](#).

De forma alternativa, puede configurar un clúster de gestores de colas. Un *clúster* es un grupo de gestores de colas configurados de forma que puedan comunicarse directamente entre ellos a través de una sola red, sin definiciones complejas de colas de transmisión, canales y colas. Los clústeres pueden configurarse con facilidad y, normalmente, contienen gestores de colas que están relacionados lógicamente de algún modo y necesitan compartir datos o aplicaciones. Incluso los clústeres más pequeños reducen los costes de administración del sistema.

El establecimiento de una red de gestores de colas en un clúster requiere menos definiciones que el establecimiento de un entorno de gestión de colas distribuidas tradicional. Como es necesario efectuar menos definiciones, la red puede configurarse o modificarse más rápida y fácilmente y el riesgo de cometer errores en las definiciones disminuye.

Para configurar un clúster, se necesita una definición de clúster emisor (CLUSDR) y una definición de clúster receptor (CLUSRCVR) para cada gestor de colas. No se necesita ninguna definición de cola de transmisión ni de cola remota. Los principios de la administración remota son los mismos cuando se utilizan dentro de un clúster, pero las definiciones propiamente dichas son significativamente más sencillas.

Si desea más información sobre cómo configurar un clúster, consulte [Configuración de un clúster de gestores de colas](#).

## Procedimiento

- Para obtener información sobre cómo administrar objetos remotos de IBM MQ, consulte los subtemas siguientes:
  - [“Configuración de gestores de colas para la administración remota”](#) en la página 208
  - [“Gestión del servidor de mandatos para la administración remota”](#) en la página 212
  - [“Emisión de mandatos MQSC en un gestor de colas remoto”](#) en la página 213
  - [“Conversión de datos entre juegos de caracteres codificados”](#) en la página 215

## Configuración de gestores de colas para la administración remota

Puede administrar un gestor de colas remoto desde un gestor de colas local utilizando los mandatos administrativos REST API, MQSC o PCF. El gestor de colas remoto podría estar en el mismo sistema, en una instalación diferente, o en un sistema distinto con el mismo entorno, o un entorno diferente de IBM MQ. Antes de poder administrar un gestor de colas de forma remota desde un gestor de colas local, debe crear un canal emisor y receptor, un escucha y una cola de transmisión en cada gestor de colas. Estos canales y colas permiten que los mandatos se envíen al gestor de colas remoto y que las respuestas se reciban en el gestor de colas local. El procedimiento para crear estas colas y canales es el mismo si desea utilizar los mandatos administrativos REST API, MQSC o PCF.

### Antes de empezar

- El procedimiento siguiente utiliza los gestores de colas de ejemplo `source.queue.manager` y `target.queue.manager`. Debe crear e iniciar estos gestores de colas en el sistema para seguir estos pasos, o sustituir sus propios nombres de gestor de colas en los pasos relevantes.
- El procedimiento siguiente utiliza TCP/IP como tipo de transporte. Debe saber la dirección IP de ambos sistemas para completar esta tarea.
- El procedimiento siguiente crea escuchas que utilizan los puertos de red 1818 en el sistema local y 1819 en el sistema remoto. Puede utilizar otros puertos, pero debe sustituir los valores de puerto en los pasos apropiados.
- Debe ejecutar los mandatos en el procedimiento localmente o a través de un recurso de red como, por ejemplo, Telnet.



## Acerca de esta tarea

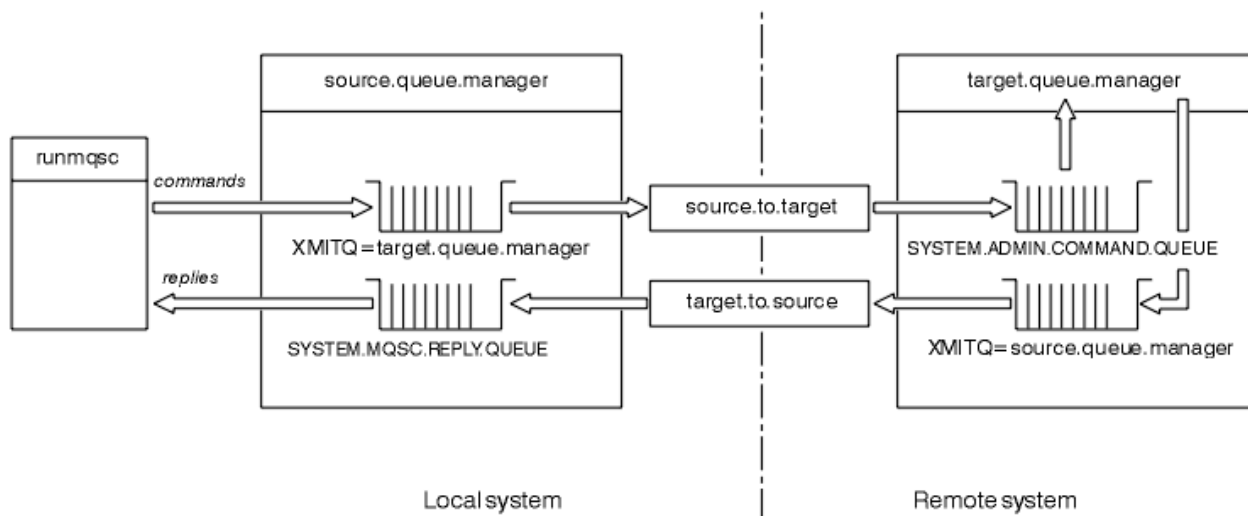


Figura 15. Preparación de canales y colas para la administración remota

Figura 15 en la página 209 muestra la configuración gestores de colas, colas y canales que necesita para la administración remota.

- El objeto `source.queue.manager` es el gestor de colas de origen desde el cual puede emitir mandatos administrative REST API, MQSC o PCF y al que se devuelven los resultados de estos mandatos.
- El objeto `target.queue.manager` es el nombre del gestor de colas de destino, que procesa los mandatos y genera todos los mensajes de operador.
- Los mandatos se colocan en la cola de transmisión que tiene el mismo nombre que el gestor de colas remoto. En este caso, `target.queue.manager`. Una cola de transmisión es una cola local especializada que contiene temporalmente mensajes antes de que el MCA los extraiga y los envíe al gestor de colas remoto.
- Los mandatos se envían mediante el canal `source.to.target` a la `SYSTEM.ADMIN.COMMAND.QUEUE` en el gestor de colas remoto. Cada extremo del canal tiene una definición distinta. Un extremo es un emisor y el otro es un receptor. Las dos definiciones deben tener el mismo nombre y, juntas, formar un solo canal de mensajes.
- La salida del mandato en la cola de transmisión remota que tiene el mismo nombre que el gestor de colas local desde el cual se envió el mandato. En este caso, `source.queue.manager`.
- La salida se envía a través del canal `target.to.source` a una cola de respuesta apropiada, donde se toma y el mandato original genera la salida.

## Procedimiento

1. En el gestor de colas del sistema remoto, asegúrese de que la cola del mandato `SYSTEM.ADMIN.COMMAND.QUEUE` está presente. Esta cola se crea de forma predeterminada cuando se crea un gestor de colas.
2. En el sistema remoto, compruebe que el servidor de mandatos se está ejecutando en el gestor de colas. Si el servidor de mandatos no se está ejecutando, la administración remota no es posible.
  - a) Inicie `runmqsc` para el gestor de colas. Por ejemplo, para el gestor de colas `target.queue.manager`, especifique el mandato siguiente:

```
runmqsc target.queue.manager
```

- b) Muestre el estado del servidor de mandatos especificando el mandato siguiente:

```
DISPLAY QMSTATUS CMDSERV
```

- c) Salga del mandato **runmqsc** especificando el mandato siguiente:

```
end
```

- d) Si el servidor de mandatos no se ha iniciado, inícielo. Por ejemplo, para el gestor de colas `target.queue.manager`, especifique el mandato siguiente:

```
strmqcsv target.queue.manager
```

3. Defina los canales, el escucha y la cola de transmisión en el gestor de colas local:

- a) Inicie **runmqsc** para el gestor de colas. Por ejemplo, para el gestor de colas `source.queue.manager`, especifique el mandato siguiente:

```
runmqsc source.queue.manager
```

- b) Defina el canal emisor. Este canal emisor debe tener el mismo nombre que el canal receptor en el gestor de colas remoto. Por ejemplo, especifique el mandato MQSC siguiente, sustituyendo el valor para **CONNNAME** con la dirección IP para el gestor de colas remoto y el número de puerto del escucha:

```
DEFINE CHANNEL ('source.to.target') +  
CHLTYPE(SDR) +  
CONNNAME (localhost:1819) +  
XMITQ ('target.queue.manager') +  
TRPTYPE(TCP)
```

- c) Defina el canal receptor. Este canal receptor debe tener el mismo nombre que el canal emisor en el gestor de colas remoto. Por ejemplo, especifique el mandato siguiente:

```
DEFINE CHANNEL ('target.to.source') +  
CHLTYPE(RCVR) +  
TRPTYPE(TCP)
```

- d) Defina el escucha en el gestor de colas local. Por ejemplo, especifique el mandato siguiente:

```
DEFINE LISTENER ('source.queue.manager') +  
TRPTYPE (TCP) +  
PORT (1818)
```

- e) Defina la cola de transmisión en el gestor de colas local. Esta cola de transmisión debe tener el mismo nombre que el gestor de colas remoto. Por ejemplo, especifique el mandato siguiente:

```
DEFINE QLOCAL ('target.queue.manager') +  
USAGE (XMITQ)
```

- f) Inicie el escucha. Por ejemplo, especifique el mandato siguiente:

```
START LISTENER ('source.queue.manager')
```

- g) Salga del mandato **runmqsc** especificando el mandato siguiente:

```
end
```

4. Defina los canales, el escucha y la cola de transmisión en el gestor de colas remoto:

- a) Inicie **runmqsc** para el gestor de colas. Por ejemplo, para el gestor de colas `target.queue.manager`, especifique el mandato siguiente:

```
runmqsc target.queue.manager
```

- b) Defina el canal emisor. Este canal emisor debe tener el mismo nombre que el canal receptor en el gestor de colas local. Por ejemplo, especifique el mandato MQSC siguiente, sustituyendo el valor para **CONNNAME** con la dirección IP para el gestor de colas local y el número de puerto para el escucha:

```
DEFINE CHANNEL ('target.to.source') +
CHLTYPE(SDR) +
CONNNAME (localhost:1818) +
XMITQ ('source.queue.manager') +
TRPTYPE(TCP)
```

- c) Defina el canal receptor. Este canal receptor debe tener el mismo nombre que el canal emisor en el gestor de colas local. Por ejemplo, especifique el mandato siguiente:

```
DEFINE CHANNEL ('source.to.target') +
CHLTYPE(RCVR) +
TRPTYPE(TCP)
```

- d) Defina el escucha. Por ejemplo, especifique el mandato siguiente:

```
DEFINE LISTENER ('target.queue.manager') +
TRPTYPE (TCP) +
PORT (1819)
```

- e) Defina la cola de transmisión. Esta cola de transmisión debe tener el mismo nombre que el gestor de colas local. Por ejemplo, especifique el mandato siguiente:

```
DEFINE QLOCAL ('source.queue.manager') +
USAGE (XMITQ)
```

- f) Inicie el escucha. Por ejemplo, especifique el mandato siguiente:

```
START LISTENER ('target.queue.manager')
```

- g) Salga del mandato **runmqsc** especificando el mandato siguiente:

```
end
```

## 5. Inicie el canal emisor en el sistema local.

- a) Inicie **runmqsc** para el gestor de colas. Por ejemplo, para el gestor de colas `source.queue.manager`, especifique el mandato siguiente:

```
runmqsc source.queue.manager
```

- b) Inicie el canal emisor. Por ejemplo, especifique el mandato siguiente:

```
START CHANNEL ('source.to.target')
```

- c) Salga del mandato **runmqsc** especificando el mandato siguiente:

```
end
```

## 6. Inicie el canal emisor en el sistema remoto.

- a) Inicie **runmqsc** para el gestor de colas. Por ejemplo, para el gestor de colas `target.queue.manager`, especifique el mandato siguiente:

```
runmqsc target.queue.manager
```

- b) Inicie el canal emisor. Por ejemplo, especifique el mandato siguiente:

```
START CHANNEL ('target.to.source')
```

- c) Salga del mandato **runmqsc** especificando el mandato siguiente:

```
end
```

7. Pruebe que la configuración se ha completado correctamente enviando un mandato MQSC desde el sistema local al gestor de colas remoto:

- a) Inicie **runmqsc** para el gestor de colas remoto desde el sistema local. Por ejemplo, especifique el mandato siguiente:

```
runmqsc -w 30 -m source.queue.manager target.queue.manager
```

- b) Muestre las colas en el gestor de colas remoto especificando el mandato siguiente:

```
DISPLAY QUEUE (*)
```

En caso de éxito, se muestra una lista de colas del gestor de colas remoto.

- c) Si estos pasos no funcionan, compruebe que los canales en ambos sistemas están en un estado de ejecución. Si los canales no se están ejecutando y no se inician, compruebe que los canales y las colas de transmisión se han configurado correctamente y que el servidor de mandatos se está ejecutando. Por ejemplo, compruebe que se ha especificado el CONNAME correcto para los canales emisores y que las colas de transmisión tienen los nombres correctos. Asimismo, consulte los registros del gestor de colas para ver excepciones de seguridad que podrían ayudar a resolver el problema.

## Resultados

Los gestores de colas se han configurado para administrar de forma remota el gestor de colas remoto desde el sistema local.

## Qué hacer a continuación

- Obtenga más información sobre la administración remota utilizando mandatos MQSC: [“Emisión de mandatos MQSC en un gestor de colas remoto”](#) en la página 213
- Obtenga más información sobre cómo escribir programas de administración utilizando mandatos PCF: [“Utilización de los formatos de mandato programable de IBM MQ”](#) en la página 26.
- Obtenga más información sobre cómo utilizar la administrative REST API para la administración remota: [“Administración remota mediante REST API”](#) en la página 80.

## Gestión del servidor de mandatos para la administración remota

Cada gestor de colas tiene un servidor de mandatos asociado. El servidor de mandatos procesa todos los mandatos entrantes procedentes de gestores de colas remotos o los mandatos PCF procedentes de aplicaciones. Presenta los mandatos al gestor de colas para su proceso y devuelve un código de terminación o un mensaje de operador. Puede iniciar, detener y mostrar el estado del servidor de mandatos. Es indispensable tener un servidor de mandatos para toda la administración que implique mandatos PCF, la MQAI y también para la administración remota.

### Antes de empezar

En función del valor del atributo del gestor de colas, **SCMDSERV**, el servidor de mandatos se inicia automáticamente cuando se inicia el gestor de colas o debe iniciarse manualmente. Si el servidor de mandatos se inicia automáticamente, no puede utilizar los mandatos **strmqcsv** o **endmqcsv** para iniciar y detener el servidor de mandatos. Puede cambiar el valor del atributo **SCMDSERV** utilizando el mandato MQSC **ALTER QMGR**. Por omisión, el servidor de mandatos se inicia automáticamente.

Al detener un gestor de colas también se detiene el servidor de mandatos asociado al mismo.

## Procedimiento

- Muestre el estado del servidor de mandatos.
  - a) Inicie **runmqsc** para el gestor de colas apropiado especificando el mandato siguiente:

```
runmqsc target.queue.manager
```

donde `target.queue.manager` es el gestor de colas para el cual se está mostrando el servidor de mandatos.

- b) Visualice el estado del servidor de mandatos especificando el mandato MQSC siguiente:

```
DISPLAY QMSTATUS CMDSERV
```

- c) Salga del mandato **runmqsc** especificando el mandato siguiente:

```
end
```

- Si el servidor de mandatos no está establecido para iniciarse automáticamente, inicie el servidor de mandatos especificando el mandato siguiente:

```
strmqcsv target.queue.manager
```

donde `target.queue.manager` es el gestor de colas para el cual se está iniciando el servidor de mandatos.

- Si el servidor de mandatos no está establecido para iniciarse automáticamente, detenga el servidor de mandatos especificando el mandato siguiente:

```
endmqcsv target.queue.manager
```

donde `target.queue.manager` es el gestor de colas para el cual se está deteniendo el servidor de mandatos.

De forma predeterminada, el servidor de mandatos se detiene de una forma controlada. Puede detener el servidor de mandatos inmediatamente añadiendo el código `-i` al mandato.

## Emisión de mandatos MQSC en un gestor de colas remoto

Después de configurar gestores de colas para la administración remota, puede utilizar un formato concreto del mandato **runmqsc** en un sistema local para ejecutar mandatos MQSC en un gestor de colas remoto. Cada mandato se envía como un PCF de escape a la cola de mandatos, `SYSTEM.ADMIN.COMMAND.QUEUE`, del gestor de colas remoto. Se reciben respuestas en la cola `SYSTEM.MQSC.REPLY.QUEUE`.

### Antes de empezar

Debe completar los pasos en [“Configuración de gestores de colas para la administración remota”](#) en la [página 208](#) para configurar canales, colas de transmisión y el servidor de mandatos antes de poder administrar remotamente un gestor de colas utilizando mandatos MQSC.

## Procedimiento

1. Asegúrese de que el servidor de mandatos se está ejecutando en el gestor de colas remoto.

Para obtener más información sobre cómo iniciar el servidor de mandatos en un gestor de colas, consulte el apartado [“Gestión del servidor de mandatos para la administración remota”](#) en la [página 212](#).
2. En el gestor de colas de origen, puede ejecutar mandatos MQSC de una de estas dos maneras:
  - De forma interactiva, iniciando **runmqsc** con los mandatos siguientes:

- **z/OS** Si el gestor de colas remoto está en z/OS, especifique el mandato siguiente:

```
runmqsc -w 30 -x -m source.queue.manager target.queue.manager
```

- **Multi** Si el gestor de colas remoto no está en z/OS, especifique el mandato siguiente:

```
runmqsc -w 30 -m source.queue.manager target.queue.manager
```

- Desde una línea de mandatos:
  - a. Coloque los mandatos MQSC que se van a ejecutar en el sistema remoto en un archivo de texto, un mandato por línea.
  - b. Verifique los mandatos MQSC en el gestor de colas local utilizando el código -v en el mandato **runmqsc**. El código -v comprueba que los mandatos son válidos, pero no los ejecuta. Tenga en cuenta que algunos mandatos podrían fallar si son aplicables al gestor de colas remoto, pero no son aplicables al gestor de colas local.

```
runmqsc -v source.queue.manager < myCmdFile.in > results.out
```

`myCmdFile.in` contiene los mandatos MQSC que se deben comprobar y el archivo `results.out` contiene los resultados de verificación de los mandatos.

- c. Ejecute el archivo de mandatos en el gestor de colas remoto especificando uno de los mandatos siguientes:

- **z/OS** Si el gestor de colas remoto está en z/OS, especifique el mandato siguiente:

```
runmqsc -w 30 -x -m source.queue.manager target.queue.manager < myCmdFile.in >  
results.out
```

- **Multi** Si el gestor de colas remoto no está en z/OS, especifique el mandato siguiente:

```
runmqsc -w 30 -m source.queue.manager target.queue.manager < myCmdFile.in >  
results.out
```

Los parámetros utilizados son los parámetros siguientes:

#### **-w segundos**

Especifica que los mandatos MQSC se ejecutan en la modalidad directa, donde los mandatos se colocan en la cola de entrada del servidor de mandatos y se ejecutan en orden.

La variable *segundos* especifica el periodo de tiempo de espera, en segundos, de una respuesta del gestor de colas remoto. Las respuestas recibidas después de este periodo de tiempo se descartan, pero los mandatos MQSC se siguen ejecutando en el gestor de colas remoto. El mensaje siguiente se genera en el gestor de colas local cuando el mandato excede el tiempo de espera:

```
AMQ8416: MQSC timed out waiting for a response from the command server.
```

Cuando acabe de emitir mandatos MQSC, el gestor de colas local mostrará cualquier respuesta con tiempo de espera excedido que haya llegado y descartará todas las respuestas posteriores.

#### **-x**

Especifica que el gestor de colas remoto es un gestor de colas z/OS.

#### **-m NombreGestorColasLocal**

Especifica el nombre del gestor de colas local que desea utilizar para enviar mandatos al gestor de colas remoto.

## Qué hacer a continuación

Si tiene dificultades para ejecutar mandatos MQSC de forma remota:

- Compruebe que se está ejecutando el gestor de colas remoto.
- Compruebe que el servidor de mandatos se está ejecutando en el sistema remoto.
- Compruebe que el intervalo de desconexión del canal no ha caducado. Por ejemplo, si se ha iniciado un canal, pero se ha concluido al cabo de un tiempo. Esto es especialmente importante si inicia manualmente los canales.
- Asegúrese de que las solicitudes que se envían desde el gestor de colas local tienen sentido en el gestor de colas de destino. Por ejemplo, las solicitudes que incluyen parámetros que no están soportados en el gestor de colas remoto.
- [“Resolución de problemas con mandatos MQSC” en la página 22](#)

## Conversión de datos entre juegos de caracteres codificados

El gestor de colas puede convertir datos de mensaje en formatos definidos de IBM MQ (también conocidos como formatos incorporados) de un juego de caracteres codificado a otro, siempre que ambos juegos de caracteres estén relacionados con un único idioma o un grupo de idiomas similares.

Por ejemplo, está soportada la conversión entre los juegos de caracteres codificados con los identificadores (CCSID) 850 y 500, porque ambos se aplican a idiomas de Europa Occidental.

Para las conversiones de caracteres de (NL) de nueva línea de EBCDIC a ASCII, consulte [Todos los gestores de colas](#).

Las conversiones soportadas están definidas en [Procesamiento de conversión de datos](#).

**V 9.1.3** A partir de IBM MQ 9.1.3, la conversión entre los CCSID 37 y 500 se admite en IBM MQ Appliance, Windows, Linux y MacOS.

## Cuando un gestor de colas no puede convertir mensajes en formatos incorporados

El gestor de colas no puede convertir automáticamente mensajes en formatos incorporados si sus CCSID respectivos representan grupos de idiomas nacionales distintos. Por ejemplo, la conversión entre el CCSID 850 y el CCSID 1025 (que es un juego de caracteres codificado EBCDIC para idiomas que utilizan el alfabeto cirílico) no está soportada porque muchos de los caracteres de uno de los juegos de caracteres codificados no pueden representarse en el otro. Si tiene una red de gestores de colas que trabajan en distintos idiomas nacionales y la conversión de datos entre algunos de los juegos de caracteres codificados no está soportada, puede habilitar una conversión predeterminada.

Para las plataformas a las que se aplica `ccsid_part2.tbl`, consulte [“Especificar la conversión de datos predeterminada” en la página 219 Utilización de `ccsid\_part2.tbl` para obtener más información](#). La conversión de datos predeterminada en plataformas distintas de las a las que se aplica el archivo `ccsid_part2.tbl` se describe en [“Conversión de datos predeterminada” en la página 217](#).

## Soporte de conversión de datos Unicode ampliado

Antes de IBM MQ 9.0, las versiones anteriores del producto no admitían la conversión de datos que contenían puntos de código Unicode más allá del plano multilingüe básico (puntos de código por encima de U+FFFF). El soporte para la conversión de datos Unicode estaba limitado a los puntos de código definidos en el estándar Unicode 3.0, codificados en UTF-8 o UCS-2, un subconjunto de anchura fija de 2 bytes de UTF-16.

A partir de IBM MQ 9.0, IBM MQ admite todos los caracteres Unicode definidos en el estándar Unicode 8.0 en la conversión de datos. Esto incluye soporte completo para UTF-16, incluyendo pares de sustitución (un par de caracteres UTF-16 de 2 bytes en el rango de X'D800' a X'DFFF' que representa un punto de código Unicode por encima de U+FFFF).

También se admite la combinación de secuencias de caracteres en casos en los que se correlacione un carácter compuesto anteriormente en un CCSID con una secuencia de caracteres de combinación en otro CCSID.

La conversión de datos a y desde Unicode y los CCSID 1388, 1390, 1399, 4933, 5488 y 16884 se ha ampliado en algunas plataformas para dar soporte a todos los puntos de código definidos actualmente para estos CCSID, incluyendo aquellos que se correlacionan con puntos de código en planos suplementarios de Unicode.

En el caso de los CCSID 1390, 1399 y 16884, esto incluye caracteres definidos en el estándar JIS X 0213 (JIS2004).

También se ha añadido soporte para la conversión a y desde Unicode y seis nuevos CCSID (de 1374 a 1379).




## Archivo `ccsid_part2.tbl`


Desde IBM MQ 9.0 se proporciona un archivo adicional, `ccsid_part2.tbl`.



El archivo `ccsid_part2.tbl` tiene prioridad sobre el archivo `ccsid.tbl` y:

- Permite añadir o modificar entradas CCSID
- Especificar la conversión de datos predeterminada
- Especificar datos para diferentes niveles de mandatos

`ccsid_part2.tbl` sólo es aplicable a las plataformas siguientes:

-  Linux - todas las versiones
-  Solaris
-  Windows

 En IBM MQ for Windows, `ccsid_part2.tbl` se encuentra en el directorio `MQDataRoot\conv\table` de forma predeterminada. Además, en IBM MQ for Windows, registra todos los conjuntos de códigos soportados.


  En las plataformas IBM MQ for Linux y Solaris, `ccsid_part2.tbl` se encuentra en el directorio `MQDataRoot/conv/table`. Para todas las plataformas Linux y Solaris, los conjuntos de códigos soportados se mantienen en tablas de conversión proporcionadas por IBM MQ.

Aunque el archivo `ccsid_part2.tbl` sustituye al archivo `ccsid.tbl` existente en versiones anteriores de IBM MQ para proporcionar información de CCSID adicional, el archivo `ccsid.tbl` sigue siendo analizado por IBM MQ y, por tanto, no se debe suprimir.

Para obtener más información, consulte [“El archivo `ccsid\_part2.tbl`” en la página 217](#).

## Archivo `ccsid.tbl`

En plataformas distintas de aquellas a las que se aplica `ccsid_part2.tbl`, el archivo `ccsid.tbl` se utiliza para los fines siguientes:

-  En AIX, el sistema operativo mantiene el conjunto de códigos soportado internamente.
- Especifica cualquier conjunto de códigos adicional. Para especificar conjuntos de códigos adicionales, debe editar `ccsid.tbl` (en el archivo se proporciona orientación sobre cómo hacerlo).
- Especifica cualquier conversión de datos predeterminada.

Puede actualizar la información registrada en `ccsid.tbl`; es posible que desee hacerlo si, por ejemplo, un release futuro del sistema operativo da soporte a conjuntos de caracteres codificados adicionales.



## Conversión de datos predeterminada

Desde IBM MQ 9.0, el método de conversión de datos predeterminada ha cambiado en las plataformas siguientes:

- Linux - todas las versiones
- Solaris
- Windows

Consulte [“Especificar la conversión de datos predeterminada”](#) en la [página 219](#) utilizando `ccsid_part2.tbl` para obtener más información.

Si configura canales entre dos máquinas en las que no se da soporte normalmente a la conversión de datos, deberá habilitar la conversión de datos predeterminada para que funcionen los canales.

En plataformas distintas de aquellas a las que se aplica `ccsid_part2.tbl`, para habilitar la conversión de datos predeterminada, edite el archivo `ccsid.tbl` para especificar un CCSID EBCDIC predeterminado y un CCSID ASCII predeterminado. Las instrucciones correspondientes están incluidas en el archivo. Debe hacer esto en todas las máquinas que se conectarán utilizando los canales. Reinicie el gestor de colas para que el cambio surta efecto.

El proceso de conversión de datos predeterminada es el siguiente:

- Si la conversión entre los CCSID de origen y de destino no está soportada, pero los CCSID de los entornos de origen y de destino son ambos EBCDIC o son ambos ASCII, los datos de tipo carácter se pasan a la aplicación de destino sin realizar ninguna conversión.
- Si un CCSID representa un juego de caracteres codificado ASCII, y el otro representa un juego de caracteres codificado EBCDIC, IBM MQ convierte los datos utilizando los CCSID de conversión de datos predeterminados definidos en `ccsid.tbl`.

**Nota:** Intente limitar los caracteres que se convierten a aquellos que tengan los mismos valores de código en el juego de caracteres codificado especificado para el mensaje y en el juego de caracteres codificado predeterminado. Si utiliza sólo el conjunto de caracteres que es válido para los nombres de objeto de IBM MQ (tal como se define en [Denominación de objetos de IBM MQ](#)) en general, cumplirá este requisito. Se producen excepciones con los CCSID EBCDIC 290, 930, 1279 y 5026 utilizados en Japón, ya que los caracteres en minúsculas tienen códigos distintos de los utilizados en otros CCSID EBCDIC.

## Conversión de mensajes en formatos definidos por el usuario

El gestor de colas no puede convertir mensajes en formatos definidos por el usuario de un juego de caracteres codificado a otro. Si necesita convertir datos en un formato definido por el usuario, debe facilitar una salida de conversión de datos para cada formato de este tipo. No utilice identificadores CCSID predeterminados para convertir datos de tipo carácter en formatos definidos por el usuario. Si desea más información sobre cómo convertir los datos en formatos definidos por el usuario y sobre cómo escribir las salidas de la conversión de datos, consulte [Escritura de salidas de conversión de datos](#).

## Cambio del CCSID del gestor de colas

Cuando haya utilizado el atributo CCSID del mandato ALTER QMGR para cambiar el CCSID del gestor de colas, detenga y reinicie el gestor de colas para asegurarse de que todas las aplicaciones que estén en ejecución, incluyendo el servidor de mandatos y los programas de canal, se detengan y se reinicien.

Esto es necesario porque cualquier aplicación que esté en ejecución cuando se cambia el CCSID del gestor de colas sigue utilizando el CCSID existente.

## El archivo `ccsid_part2.tbl`

El archivo `ccsid_part2.tbl` se utiliza para proporcionar información CCSID adicional. El archivo `ccsid_part2.tbl` sustituye el archivo `ccsid.tbl` que se ha utilizado antes de IBM MQ 9.0.

**Nota:** El archivo `ccsid.tbl`, que se utilizaba antes de IBM MQ 9.0 para suministrar información de CCSID adicional, continúa siendo analizado por IBM MQ y no debe suprimirse. Sin embargo, las entradas en `ccsid_part2.tbl` tienen prioridad sobre otras entradas de `ccsid.tbl`.




Deberá utilizar `ccsid_part2.tbl` en lugar de `ccsid.tbl` porque `ccsid_part2.tbl`:

- Contiene soporte para los valores de codificación Unicode. Desde IBM MQ 9.0, IBM MQ admite todos los caracteres Unicode definidos en el estándar Unicode 8.0 en conversión de datos, incluido soporte completo para UTF-16. Para obtener más información, consulte [“Conversión de datos entre juegos de caracteres codificados”](#) en la página 215.
- Le permite especificar la versión de las entradas de CCSID, para que las entradas sólo sean aplicables a los niveles de mandatos seleccionados.




Puede utilizar el archivo `ccsid_part2.tbl` para:

- Añadir o modificar entradas CCSID
- Especificar la conversión de datos predeterminada
- Especificar datos para diferentes niveles de mandatos

El archivo `ccsid_part2.tbl` sólo es aplicable a las plataformas siguientes:

-  Linux - todas las versiones
-  Solaris
-  Windows

La ubicación del archivo `ccsid_part2.tbl` depende de la plataforma:

-   El directorio `MQDataRoot/conv/table` en todas las versiones de Linux y en Solaris.
-  El directorio `MQDataRoot\conv\table` en Windows.

## Adición o modificación de entradas de CCSID

Una entrada del archivo `ccsid_part2.tbl` tiene el formato siguiente:

```
<CCSID number> <Base CCSID> <DBCS CodePage> <SBCS CodePage>  
<Type> <Encoding> <ACRI> <Name>
```

Una entrada de ejemplo para el CCSID 1200 (UTF-16) es:

```
1200 1200 1200 1200 3 8 0 UTF-16
```

**Nota:** Para obtener más detalles sobre el valor para ACRI, consulte el comentario del archivo `ccsid_part2.tbl`.

En el formato de `ccsid_part2.tbl`:

### El tipo puede ser igual a:

- 1=SBCS
- 2=DBCS
- 3=MBCS

### El cifrado puede ser igual a:

- 1=EBCDIC
- 2 = ASCII
- 3 = ISO
- 4 = UCS-2

- 5 = UTF-8
- 6 = Euc
- 7 = GB18030
- 8 = UTF-16
- 9 = UTF-32

Al editar el archivo:

- Puede especificar un comentario utilizando el símbolo # al comienzo de una línea. Esto evita que IBM MQ intente analizar la línea.
- No puede proporcionar comentarios en la misma línea.
- Debe asegurarse de que no crea líneas en blanco.
- No debe añadir nuevas entrada al final del archivo.

Las nuevas entradas de CCSID deben añadirse antes de la información de tabla de ACRI.

## Especificar la conversión de datos predeterminada

Puede definir los CCSID de conversión predeterminados, que se utilizan para convertir entre CCSID ASCII o similares y EBCDIC, si no se admite ninguna conversión entre dos CCSID.

Si se habilita esta función, se usa la conversión predeterminada en las cabeceras de mensajes y transmisión, y también se puede utilizar en la conversión de datos de usuario.

Las conversiones predeterminadas se habilitan mediante la creación de dos líneas similares a las siguientes:

```
default      0      500      1      1      0
default      0      850      1      2      0
```

La primera línea establece el valor predeterminado para CCSID EBCDIC en 500 y la segunda línea establece el valor predeterminado para CCSID ASCII y similares en 850.

## Especificación de datos para distintos niveles de mandatos

Para especificar entradas de CCSID para distintos niveles de mandatos de IBM MQ, utilice un símbolo de dos puntos seguido por el nivel de mandatos (o niveles de mandatos) de IBM MQ al que desee que se aplique la siguiente sección.

El número representa el nivel de mandatos mínimo en el que debe ejecutarse el gestor de colas o el cliente. Por ejemplo, si el gestor de colas actual está en el nivel de mandatos 900, y detecta un distintivo de nivel de mandatos 800 o 900, se leen los CCSID.

No obstante, un gestor de colas a nivel 800 pasará por alto los CCSID de la sección 900.

El nivel de mandatos especificado se aplica a todas las entradas CCSID encontradas tras un distintivo de nivel de mandatos, hasta que se encuentre un nuevo distintivo de nivel de mandatos.

Si necesita establecer el nivel de mandatos en todos los niveles de mandatos, especifique el número cero.

Al analizar `ccsid_part2.tbl` por primera vez, IBM MQ trata todos los CCSID encontrados como válidos para todos los niveles de mandatos de IBM MQ.

Se empiezan a utilizar versiones únicamente cuando IBM MQ encuentra el primer distintivo de nivel de mandatos.

El fragmento de código siguiente muestra un ejemplo que utiliza versiones:

```
# Comment Block
# End of Comment Block
# Because no command level flag is specified and we're at the start of the file
# the following CCSIDs will be read on all versions
```

```

819 819 0 819 1 3 0 ISO8859-1
923 923 0 923 1 3 0 ISO8859-15
1051 1051 0 1051 1 3 0 IBM-1051
# The colon :900 below shows that the CCSIDs after will only be for MQ cmd level 900 and above
:900
8629 437 0 437 1 2 0 IBM-437
12725 437 0 437 1 2 0 IBM-437
16821 437 0 437 1 2 0 IBM-437
20917 437 0 437 1 2 0 IBM-437
# The colon :0 below shows that the CCSIDs after will be for all version of MQ
:0
4946 850 0 850 1 2 0 IBM-850
33618 850 0 850 1 2 0 IBM-850
61697 850 0 850 1 2 0 IBM-850
61698 850 0 850 1 2 0 IBM-850

```

## Administración de Managed File Transfer

Utilice los mandatos de Managed File Transfer para administrar Managed File Transfer. También puede utilizar IBM MQ Explorer para realizar algunas de las tareas administrativas.

### Inicie la transferencia colocando un mensaje en una cola de mandatos de agente

También puede iniciar una transferencia de archivos colocando un mensaje de transferencia de archivos en la cola de mandatos del agente de origen. Un nombre de cola de mandatos de ejemplo es `SYSTEM.FTE.COMMAND.AGENT01`. Debe asegurarse de que el mensaje llegue a la cola de mandatos del agente de origen correcto; si el mensaje lo recibe un agente que no coincide con la información de origen del XML, el mensaje es rechazado.


La solicitud de transferencia XML debe ajustarse al esquema `FileTransfer.xsd` y utilizar el elemento `<request>` como elemento raíz. Para obtener información sobre la estructura y el contenido de un mensaje de petición de transferencia, consulte [Formato de mensaje de petición de transferencia de archivos](#). El modo de colocar el mensaje de solicitud de transferencia en la cola de mandatos de un agente depende de la tarea. Por ejemplo, puede utilizar la API IBM MQ Java para colocar un mensaje en la cola de forma programática.


## Iniciar un agente de MFT

Para poder utilizar un agente de Managed File Transfer para una transferencia de archivos, antes debe iniciar el agente.

### Acerca de esta tarea

Puede iniciar un Managed File Transfer Agent desde la línea de mandatos. En este caso, el proceso del agente se detiene cuando finaliza sesión en el sistema.

 En UNIX, Linux, and Windows, puede configurar un agente para que se continúe ejecutando cuando cierre sesión del sistema y pueda continuar recibiendo las transferencias de archivos.

 En z/OS, puede configurar el agente para que se inicie como una tarea inicia desde JCL, sin necesidad de una sesión interactiva.

Tenga en cuenta que si un agente encuentra un error irreparable cuando se está ejecutando, se genera una primera FDC (Failure Data Capture) y se detiene el agente.

### Procedimiento

- Para iniciar un agente desde la línea de mandatos, utilice el mandato **`fteStartAgent`**. Para obtener más información, consulte [fteStartAgent](#).



- Para configurar un agente, de modo que se continúe ejecutando cuando finaliza sesión en el sistema:

- **Windows** En Windows, configure el agente para que se ejecute como un servicio de Windows. Para obtener más información, consulte [“Inicio de un agente MFT como un servicio de Windows”](#) en la página 221.
- **Linux** **UNIX** En UNIX y en Linux, configure el agente para que se inicie automáticamente durante un reinicio utilizando un archivo de script. Para obtener más información, consulte [“Inicio de un agente MFT durante el arranque del sistema UNIX y Linux”](#) en la página 222.
- **z/OS**
  - En z/OS, configure el agente para que se inicie como una tarea iniciada desde JCL sin necesidad de una sesión interactiva.
  - Para obtener más información, consulte [“Inicio de un agente MFT en z/OS”](#) en la página 224.

## **Windows** Inicio de un agente MFT como un servicio de Windows

Puede iniciar un agente como un servicio de Windows, de modo que cuando finalice la sesión de Windows, el agente continúa ejecutándose y puede recibir transferencias de archivos.

### Acerca de esta tarea

En Windows, cuando inicia un agente desde la línea de mandatos, el proceso de agente se ejecuta utilizando el nombre de usuario que ha utilizado para iniciar la sesión en Windows. Cuando finaliza la sesión del sistema, el proceso del agente se detiene. Para impedir que el agente se detenga, puede configurar un agente para que se ejecute como un servicio de Windows. La ejecución como un servicio de Windows también permite configurar agentes para que se inicien automáticamente cuando se inicia o se reinicia el entorno de Windows.

Realice los pasos siguientes para iniciar un agente que se ejecute como un servicio de Windows. Debe estar ejecutando Managed File Transfer en una de las versiones de Windows soportadas para ejecutar el agente como un servicio de Windows. Para ver la lista de entornos soportados, consulte los [Requisitos del sistema para IBM MQ](#).

Los pasos exactos dependen de si ya ha creado un agente o de si lo está creando. Ambas opciones se describen en los pasos siguientes.

### Procedimiento

1. Si está creando un agente de Managed File Transfer, utilice el mandato **fteCreateAgent**, **fteCreateCDAgent** o **fteCreateBridgeAgent**. Especifique el parámetro **-s** para ejecutar el agente como un servicio de Windows. En el ejemplo siguiente, se crea el agente AGENT1, que tiene un gestor de colas del agente QMGR1. El servicio de Windows se ejecuta con el nombre de usuario fteuser, que tiene asociada la contraseña ftepassword.

```
fteCreateAgent -agentName AGENT1 -agentQMGR QMGR1 -s -su fteuser -sp ftepassword
```

Puede especificar opcionalmente un nombre para el servicio después del parámetro **-s**. Si no especifica un nombre, el servicio se denomina `mqmftAgentAGENTQMGR`, donde *AGENT* es el nombre de agente que ha especificado y *QMGR* es el nombre del gestor de colas de agente. En este ejemplo, el nombre predeterminado para el servicio es `mqmftAgentAGENT1QMGR1`.

**Nota:** La cuenta de usuario Windows que especifique utilizando el parámetro **-su** debe tener los derechos **Log on as a service**. Si desea información sobre cómo configurar esto, consulte [Instrucciones para ejecutar un agente o un registrador MFT como un servicio Windows](#).

Para obtener más información, consulte [fteCreateAgent](#), [fteCreateCDAgent: crear un agente de puente Connect:Direct](#) o [fteCreateBridgeAgent \(crear y configurar un agente de puente de protocolo MFT\)](#).

2. Si ha seguido el paso anterior para crear un agente, ejecute los mandatos MQSC generados por el mandato **fteCreateAgent**, **fteCreateCDAgent** o **fteCreateBridgeAgent**. Estos mandatos crean las colas de IBM MQ que el agente necesita.

Por ejemplo, para un agente denominado *AGENT1*, un gestor de colas del agente denominado *QMGR1* y un gestor de colas de coordinación denominado *COORDQMGR1*, ejecute el mandato siguiente:

```
runmqsc QMGR1 MQ_DATA_PATH\mqft\config\COORDQMGR1\agents\AGENT1\AGENT1_create.mqsc
```

3. Si no ha seguido los pasos anteriores para crear un agente y en su lugar desea configurar un agente existente para que se ejecute como servicio de Windows, detenga primero el agente si está en ejecución y, a continuación, modifique la configuración.

a) El ejemplo siguiente utiliza un agente denominado *AGENT1*. Ejecute el mandato siguiente:

```
fteStopAgent AGENT1
```

b) Utilice el mandato **fteModifyAgent** para configurar el agente para que se ejecute como un servicio de Windows:

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

Si desea más información, consulte [fteModifyAgent: ejecutar un agente MFT como un servicio Windows](#).

4. Inicie el agente utilizando el mandato **fteStartAgent**. De forma alternativa, puede utilizar la herramienta de Servicios de Windows, que está disponible en las herramientas administrativas en el Panel de control, que se selecciona desde el menú de inicio del escritorio de Windows para iniciar el servicio.

```
fteStartAgent AGENT1
```

El servicio continúa ejecutándose aunque haya finalizado la sesión en Windows. Para garantizar que el servicio también se reinicia cuando se reinicia Windows después de una conclusión, el campo **Tipo de inicio** en la herramienta Servicios de Windows se establece en **Automático** de forma predeterminada. Cámbielo por **Manual** si no desea que el servicio se reinicie cuando se reinicie Windows.

5. Opcional: Para detener el agente, utilice el mandato **fteStopAgent** o utilice la herramienta Windows Services. Por ejemplo, desde la línea de mandatos, ejecute el mandato siguiente:

```
fteStopAgent AGENT1
```

- Cuando se ejecuta el mandato **fteStopAgent** como un servicio, el mandato siempre se ejecuta utilizando el parámetro **-i**, independientemente de si se ha especificado este parámetro. El parámetro **-i** detiene el agente inmediatamente sin completar las transferencias que están en curso. Esto se debe a una limitación del servicio de Windows.

## Qué hacer a continuación

Si tiene problemas al iniciar el servicio Windows, consulte [Instrucciones para ejecutar un agente o registrador de MFT como un servicio Windows](#). En este tema también se describe la ubicación de los archivos de registro de servicio de Windows.

Linux

UNIX

## Inicio de un agente MFT durante el arranque del sistema UNIX y Linux

Un Managed File Transfer Agent puede configurarse para que se inicie durante el arranque del sistema en UNIX. Cuando finalice la sesión en un sistema UNIX, el agente continuará ejecutándose y puede recibir transferencias de archivos.

Cuando haya creado y configurado un agente utilizando uno de estos mandatos de Managed File Transfer; **fteCreateAgent**, **fteCreateCDAgent**, o **fteCreateBridgeAgent**, puede configurarlo para que se

inicie automáticamente durante un rearranque en máquinas UNIX utilizando un archivo de script que simplemente ejecute:

```
su -l mqmft_user -c mq_install_root/bin/fteStartAgent agent_name
```

Donde *mq\_install\_root* es el directorio raíz de la instalación de Managed File Transfer necesaria, el valor predeterminado es: */opt/mqm* y *nombre\_agente* es el nombre del Managed File Transfer Agent que se va a iniciar. El uso de este archivo de script varía en función del sistema operativo UNIX específico.

## Solaris

### Solaris

En Solaris, siga estos pasos:

1. Ejecute el mandato siguiente y realice un seguimiento de la vía de acceso devuelta:

```
which sh
```

Por ejemplo, la vía de acceso puede ser */usr/bin/sh*

2. Como ID de usuario root del sistema, cree su propio archivo */etc/init.d/startmqmft*.
3. Edite este archivo y añádale las líneas de script, utilizando la vía de acceso devuelta en el paso 1 como primera línea en el script:

```
#!/usr/bin/sh
su mqmft_user mq_install_root/bin/fteStartAgent agent_name
```

4. Convierta el archivo en ejecutable, por ejemplo:

```
chmod 755 /etc/init.d/startmqmft
```

5. Realice symlink del archivo al directorio *rc3.d*:

```
ln -s /etc/init.d/startmqmft /etc/rc3.d/S98startmqmft
```

El prefijo S indica el estado predeterminado Iniciado en Solaris. 98 es un número de secuencia. El sufijo es el nombre de archivo de *init.d*

## Linux

### Linux

Para los sistemas Linux hay varias formas de iniciar aplicaciones durante el proceso de arranque del sistema. En general, considere realizar los pasos siguientes:

1. Cree un archivo denominado */etc/rc.mqmft* con contenido:

```
#!/bin/sh
su -l mqmft_user"-c mq_install_root/bin/fteStartAgent agent_name"
```

Donde *mqmft\_user* es el ID de usuario bajo el que se va a ejecutar el proceso de agente. Este ID de usuario debe ser miembro del grupo *mqm*.

2. Convierta el archivo en ejecutable, por ejemplo:

```
chmod 755 /etc/rc.mqmft
```

3. A continuación, añada la siguiente línea a */etc/inittab*:

```
mqmft:5:boot:/etc/rc.mqmft
```

Otras formas de iniciar un agente durante el arranque en Linux incluye añadir las líneas de script al archivo `/etc/rc.d/rc.local` o en Linux SuSe, añadiendo las líneas de script al archivo `/etc/init.d/boot.local`. Debe seleccionar el método que funcione mejor en su entorno. A continuación se muestra más información sobre otras maneras de iniciar un agente durante el arranque en distribuciones de Linux específicas que están soportadas:

## SLES 10 y 11

En los sistemas SUSE Linux Enterprise Server (SLES) 10 y 11, siga estos pasos:

1. Como ID de usuario root del sistema, cree su propio archivo `/etc/init.d/rc.rclocal`.
2. Añada las líneas siguientes al archivo `rc.rclocal`:

```
#!/bin/sh
### BEGIN INIT INFO
# Provides: rc.rclocal
# Required-Start: $network $syslog
# Required-Stop: $network $syslog
# Default-Stop: 0 1 2 6
# Description: MQMFT agent startup
### END INIT INFO
su -l mqmft_user"-c mq_install_root/bin/fteStartAgent agent_name"
```

3. Ejecute los mandatos siguientes:

```
chmod 755 rc.rclocal
chkconfig --add rc.rclocal
```

## Inicio de un agente MFT en z/OS

En z/OS, además de ejecutar el mandato **fteStartAgent** desde una sesión de UNIX System Services, se puede iniciar un agente como una tarea iniciada desde JCL sin necesidad de una sesión interactiva.

Se utiliza una tarea iniciada porque se ejecuta bajo un ID de usuario específico y no se ve afectada por el registro de usuarios.

**Nota:** Las tareas iniciadas se suelen ejecutar bajo el usuario administrativo que puede que no tenga privilegios de inicio de sesión por lo tanto no es posible iniciar sesión en el sistema z/OS con el usuario bajo el que se ejecuta el agente. Los mandatos **fteStartAgent**, **fteStopAgent**, **fteSetAgentTraceLevel** y el mandato **fteShowAgentDetails** con el parámetro **-d** especificado no se puede emitir para dicho agente.

Desde IBM MQ 9.0.2 y IBM MQ 9.0.0 Fix Pack 1, puede utilizar la propiedad de agente **adminGroup** con agentes Managed File Transfer en z/OS. Puede definir un grupo gestor de seguridad, por ejemplo, MFTADMIN y, a continuación, añadir el ID de usuario de tarea iniciada y los ID de TSO de administrador a este grupo. Edite el archivo de propiedades de agente y establezca la propiedad **adminGroup** para que tenga el mismo nombre de este grupo gestor de seguridad.

```
adminGroup=MFTADMIN
```

Los miembros de este grupo pueden emitir los mandatos **fteStartAgent**, **fteStopAgent** y **fteSetAgentTraceLevel**, y el mandato **fteShowAgentDetails** con el parámetro **-d** especificado, para el agente que se está ejecutando como una tarea iniciada.

Si desea más información, consulte la propiedad **adminGroup** en [El archivo de MFT agent.properties](#).

Como aplicación Java, un agente es una aplicación de UNIX System Services que puede ejecutar desde JCL utilizando el miembro BFGAGSTP, desde un conjunto de datos de la biblioteca PDSE de mandatos de Managed File Transfer generado, para un agente. Si desea más información sobre cómo crear un conjunto



de datos de biblioteca de mandato PDSE de MFT y personalizarlo para el agente necesario, consulte [Creación de un conjunto de datos del mandato de agente o registrador de MFT](#).

## Habilite las conexiones remotas con IBM MQ Advanced for z/OS Value Unit Edition solo



Un agente de MFT en z/OS, que se ejecuta bajo el identificador de producto (ID de producto) de IBM MQ Advanced for z/OS VUE, se conecta a un gestor de colas en z/OS mediante una conexión de cliente.

Tenga en cuenta que el PID bajo el que se ejecuta el agente se muestra en el registro en el inicio del agente.

Para obtener detalles de los productos de IBM MQ, sus valores de PID asociados y las clasificaciones de exportación, consulte [Identificadores del producto IBM MQ e información de exportación](#).

Un agente de MFT en z/OS, que se ejecuta bajo cualquier otro ID de producto, se conecta a un gestor de colas local solo mediante conexiones de enlace.

Un agente de MFT en z/OS se conecta solo a un gestor de colas, además se ejecuta en z/OS, independientemente del ID de producto de MFT.

Si un agente de IBM MQ Advanced for z/OS VUE intenta conectarse a un gestor de colas que no se está ejecutando en z/OS, se emite el mensaje `BFGMQ1044E` siguiente y finaliza el inicio del agente:

`BFGMQ1044E: La conexión de cliente del agente en z/OS debe ser a un gestor de colas en z/OS`

### Referencia relacionada

[“Detención de un agente MFT en z/OS” en la página 226](#)

Si está ejecutando un Managed File Transfer Agent en z/OS como una tarea iniciada desde JCL, el agente acepta los z/OS mandatos de operador **MODIFY** y **STOP**, además del mandato **fteStopAgent**.

El archivo `MFT agent.properties`

## Listar agentes de MFT

Puede listar los agentes registrados Managed File Transfer con un gestor de colas específico mediante la línea de mandatos o bien mediante IBM MQ Explorer.

### Acerca de esta tarea

Para listar agentes mediante la línea de mandatos, consulte el [mandato fteListAgents](#).

Para listar agentes utilizando IBM MQ Explorer, en la vista de navegador pulse **Agentes** bajo el nombre del gestor de colas de coordinación.

Si un agente no aparece en la lista del mandato **fteListAgents** o no se muestra en IBM MQ Explorer, utilice el diagrama de flujo de diagnósticos en el tema siguiente para localizar y arreglar el problema: [¿Qué hacer si el agente MFT no está listado por el mandato fteListAgents?](#)

## Detener un agente de MFT

Puede detener un agente Managed File Transfer desde la línea de mandatos. Cuando detiene un agente, desactiva el agente y permite que el agente complete la transferencia de archivos actual antes de que se detenga. También puede especificar el parámetro **-i** en la línea de mandatos para detener inmediatamente un agente. Cuando el agente se ha detenido, no puede utilizar ese agente para transferir archivos hasta que lo reinicie.

### Antes de empezar

Si desea comprobar los nombres de los agentes asociados con un gestor de colas, puede listar los agentes utilizando IBM MQ Explorer o la línea de mandatos; consulte el [mandato fteListAgents](#).

## Acerca de esta tarea

Para detener un agente desde la línea de mandatos, consulte [fteStopAgent](#).

Si ha configurado su agente para que se ejecute como servicio de Windows, ejecutando el mandato **fteStopAgent** también se detiene el servicio de Windows. O bien, puede detener el agente deteniendo el servicio mediante la herramienta Servicios de Windows. Para obtener más información, consulte el tema “Inicio de un agente MFT como un servicio de Windows” en la página 221.

## Detención de un agente MFT en z/OS

Si está ejecutando un Managed File Transfer Agent en z/OS como una tarea iniciada desde JCL, el agente acepta los z/OS mandatos de operador **MODIFY** y **STOP**, además del mandato **fteStopAgent**.

Se utiliza una tarea iniciada porque se ejecuta bajo un ID de usuario específico y no se ve afectada por el registro de usuarios.

**Nota:** Las tareas iniciadas se suelen ejecutar bajo el usuario administrativo que puede que no tenga privilegios de inicio de sesión por lo tanto no es posible iniciar sesión en el sistema z/OS con el usuario bajo el que se ejecuta el agente. Los mandatos **fteStartAgent**, **fteStopAgent**, **fteSetAgentTraceLevel** y el mandato **fteShowAgentDetails** con el parámetro **-d** especificado no se puede emitir para dicho agente.

Desde IBM MQ 9.0.2 y IBM MQ 9.0.0 Fix Pack 1, puede utilizar la propiedad de agente **adminGroup** con agentes Managed File Transfer en z/OS. Puede definir un grupo gestor de seguridad, por ejemplo, MFTADMIN y, a continuación, añadir el ID de usuario de tarea iniciada y los ID de TSO de administrador a este grupo. Edite el archivo de propiedades de agente y establezca la propiedad **adminGroup** para que tenga el mismo nombre de este grupo gestor de seguridad.

```
adminGroup=MFTADMIN
```

Los miembros de este grupo pueden emitir los mandatos **fteStartAgent**, **fteStopAgent** y **fteSetAgentTraceLevel**, y el mandato **fteShowAgentDetails** con el parámetro **-d** especificado, para el agente que se está ejecutando como una tarea iniciada.

Si desea más información, consulte la propiedad **adminGroup** en [El archivo de MFT agent.properties](#).

## Conclusión controlada del agente utilizando el mandato MODIFY de z/OS (F)

El mandato **MODIFY** le permite detener un agente de forma controlada como alternativa al mandato **fteStopAgent**. El agente completa las transferencias actualmente en curso pero el agente no inicia ninguna nueva transferencia.

Por ejemplo:

```
F job_name,APPL=STOP
```

donde *nombre\_trabajo* es el trabajo en que se ejecuta el proceso de agente.

## Conclusión inmediata del agente utilizando el mandato STOP de z/OS (P)

El mandato **STOP** equivale a una detención inmediata utilizando el mandato **fteStopAgent** con el parámetro **-i**. El agente se detiene inmediatamente aunque el agente esté transfiriendo actualmente un archivo.

Por ejemplo:

```
P job_name
```

donde *nombre\_trabajo* es el trabajo en que se ejecuta el proceso de agente.

### Referencia relacionada

“Inicio de un agente MFT en z/OS” en la página 224

En z/OS, además de ejecutar el mandato **fteStartAgent** desde una sesión de UNIX System Services, se puede iniciar un agente como una tarea iniciada desde JCL sin necesidad de una sesión interactiva.

El archivo `MFT agent.properties`

## Inicio de una nueva transferencia de archivos

Puede iniciar una nueva de transferencia de archivos desde IBM MQ Explorer o desde la línea de mandatos y puede elegir transferir un único archivo o varios archivos de un grupo.

### Acerca de esta tarea

Para iniciar una nueva transferencia de archivos desde la línea de mandatos, consulte el [mandato fteCreateTransfer](#).

Para iniciar una nueva transferencia de archivos utilizando el asistente **Crear nueva transferencia de archivos gestionada** en IBM MQ Explorer, siga los pasos que se indican a continuación:

### Procedimiento

1. En la vista Navegador, pulse **Transferencia de archivos gestionada**. En la vista Contenido, aparece **Central de transferencias de archivos gestionadas**.
2. En la vista Navegador, aparecen todos los gestores de colas de coordinación. Expanda el nombre del gestor de colas de coordinación en el que está registrado el agente que desea utilizar para la transferencia. Si está conectado actualmente a un gestor de colas de coordinación que no sea el que desea utilizar para la transferencia, pulse el botón derecho del ratón en dicho nombre de gestor de colas de coordinación en la vista Navegador y pulse **Desconectar**. A continuación, pulse el botón derecho del ratón en el nombre del gestor de colas de coordinación que desea utilizar y pulse **Conectar**.
3. Inicie el asistente **Crear nueva transferencia de archivos gestionada** utilizando cualquiera de los dos métodos siguientes:
  - a) Pulse el botón derecho del ratón en el nombre de cualquiera de los nodos siguientes en la vista Navegador: el gestor de colas de coordinación relevante, **Plantillas de transferencia**, **Registro de transferencias** o **Transferencias pendientes**. A continuación, pulse **Nueva transferencia** para iniciar el asistente.
  - b) Pulse **Archivo > Nuevo > Otros > Asistentes de transferencia de archivos gestionada > Asistente de nueva transferencia**
4. Siga las instrucciones de los paneles del asistente. También se proporciona ayuda según contexto para cada panel. Para acceder a la ayuda según contexto en Windows, pulse F1. En Linux, pulse `Ctrl+F1` o `Shift+F1`.

## Utilización de archivos de definición de transferencia

Puede especificar un archivo de definición de transferencia para crear una transferencia de archivos. El archivo de definición de transferencia es un archivo XML que define parte o toda la información requerida para crear la transferencia.

Los archivos de definición de transferencia resultan útiles si desea especificar varios archivos de origen y varios archivos de destino en una sola operación de transferencia. Puede utilizar un archivo de definición de transferencia para someter una transferencia de archivos compleja. Puede reutilizar y compartir el archivo de definición de transferencia.

Puede utilizar dos formatos para un archivo de definición de transferencia, y aunque estos formatos varían ligeramente, ambos se ajustan al esquema de `FileTransfer.xsd`. Puede encontrar este esquema en el directorio `samples\schema` de la instalación de Managed File Transfer.

Los siguientes dos formatos de archivos de definición de transferencia están soportados:

- Una definición de los archivos de origen y destino de una transferencia. Esta definición utiliza un elemento **transferSpecifications** como raíz.
- Una definición de la transferencia completa, incluyendo archivos de origen y destino y los agentes de origen y destino. Esta definición utiliza un elemento **request** como raíz.
  - Se pueden generar archivos con este formato desde el mandato **fteCreateTransfer** utilizando el parámetro **-gt**.

El ejemplo siguiente muestra un formato de archivo de definición de transferencia que especifica únicamente los archivos de origen y destino para una transferencia:

```
<?xml version="1.0" encoding="UTF-8"?>
<transferSpecifications xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <item checksumMethod="MD5" mode="text">
    <source recursive="false" disposition="leave">
      <file>textTransferTest.txt</file>
    </source>
    <destination type="directory" exist="overwrite">
      <file>c:\targetfiles</file>
    </destination>
  </item>
</transferSpecifications>
```

Para enviar este formato de archivo de definición de transferencia deberá especificar los agentes de origen y destino en la línea de mandatos:

```
fteCreateTransfer -sa AGENT1 -sm agent1qm -da AGENT2 -dm agent2qm -td
c:\definitions\example1.xml
```

El siguiente ejemplo es un formato de archivo de definición de transferencia que especifica toda la información requerida para una transferencia:

```
<?xml version="1.0" encoding="UTF-8"?>
<request version="3.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>example.com.</hostName>
      <userID>fteuser</userID>
    </originator>
    <sourceAgent agent="AGENT1" QMgr="agent1qm"/>
    <destinationAgent agent="AGENT2" QMgr="agent2qm"/>
    <transferSet>
      <item mode="binary" checksumMethod="MD5">
        <source recursive="false" disposition="leave">
          <file>c:\sourcefiles\*.jpg</file>
        </source>
        <destination type="directory" exist="error">
          <file>/targetfiles/images</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Puede generar un archivo con este formato utilizando el parámetro **-gt** en el mandato **fteCreateTransfer**. Cuando envía un archivo de definición de transferencia con este formato, no necesita especificar nada más en la línea de mandatos:

```
fteCreateTransfer -td c:\definitions\example2.xml
```

Puede alterar temporalmente la información del agente de origen y destino en la línea de mandatos pasando los parámetros normales además del archivo de definición de transferencia. Por ejemplo:

```
fteCreateTransfer -da AGENT9 -dm agent9qm -td c:\definitions\example2.xml
```

Este ejemplo utiliza las opciones de línea de mandatos para alterar temporalmente el agente de destino definido dentro del archivo de definición de transferencia con **AGENT9** y el gestor de colas de destino definido en el archivo de definición de transferencia como **agent9qm**.

Los dos formatos descritos pueden contener uno o más elementos <item>. Para obtener más información sobre el elemento <item> consulte [Formato del mensaje de solicitud de transferencia de archivos](#). Cada uno de estos elementos de transferencia define un par de archivo de origen y destino con atributos adicionales para controlar el comportamiento de la transferencia. Por ejemplo, puede especificar el comportamiento siguiente:

- Si la transferencia utiliza una suma de comprobación
- Si la transferencia es de texto o binaria
- Si desea suprimir el archivo de origen después de que haya finalizado la transferencia
- Si desea sobrescribir el archivo de destino si el archivo existe


Una ventaja de utilizar archivos de definición de transferencia es que puede especificar opciones adicionales que no están disponibles desde la línea de mandatos. Por ejemplo, cuando esté llevando a cabo transferencias de mensaje a archivo, puede especificar el atributo groupId utilizando un archivo de definición de transferencia. Este atributo especifica el ID de grupo de IBM MQ de los mensajes que se leen de la cola. Otra ventaja de utilizar archivos de definición de transferencia es que puede especificar distintas opciones para cada par de archivos. Por ejemplo, puede especificar si se utiliza una suma de comprobación, o si el archivo se transfiere en modalidad de texto o binaria, individualmente para cada archivo. Si utiliza la línea de mandatos, se aplican las mismas opciones para cada archivo de una transferencia.


Por ejemplo:

```
<item checksumMethod="none" mode="binary">
  <source disposition="leave">
    <file>c:\sourcefiles\source1.doc</file>
  </source>
  <destination type="file" exist="error">
    <file>c:\destinationfiles\destination1.doc</file>
  </destination>
</item>

<item checksumMethod="MD5" mode="text">
  <source disposition="delete">
    <file>c:\sourcefiles\source2.txt</file>
  </source>
  <destination type="file" exist="overwrite">
    <file encoding="UTF8" EOL="CRLF">c:\destinationfiles\destination2.txt</file>
  </destination>
</item>

<item checksumMethod="none" mode="text">
  <source recursive="false" disposition="leave">
    <file>c:\originfiles\source3.txt</file>
  </source>
  <destination type="file" exist="overwrite">
    <file>c:\targetfiles\destination3.txt</file>
  </destination>
</item>
```

 Se pueden usar elementos para transferir un archivo de un sistema distribuido a un sistema z/OS:

```

<item checksumMethod="none" mode="text">
  <source recursive="false" disposition="leave">
```

```
<file>textTransferTest.txt</file>
</source>
<destination type="dataset" exist="overwrite">
  <file encoding="IBM-1047">//TEXT.TRANS.TEST</file>
</destination>
</item>
```

**z/OS** Este ejemplo transfiere el archivo `textTransferTest.txt` del agente de origen al conjunto de datos `//TEXT.TRANS.TEST` en el agente de destino en modalidad de texto. Esta transferencia convierte los datos de origen de la codificación predeterminada del agente de origen (no se especifica ningún atributo de codificación de origen) a la página de códigos: IBM-1047.

## Creación de una transferencia planificada de archivos

Puede planificar una nueva transferencia de archivos desde IBM MQ Explorer o desde la línea de mandatos. La transferencia planificada puede contener archivos individuales o varios archivos en un grupo. Puede realizar una transferencia de archivos planificada una vez o repetirla varias veces.

### Acerca de esta tarea

Puede configurar una planificación de transferencia de archivos para que se realice una vez o para que se realice en los siguientes intervalos:

- Cada minuto
- Cada hora
- Diariamente
- Semanalmente
- Mensualmente
- Anualmente

A continuación, puede especificar que las apariciones se detengan en los puntos siguientes:

- A una hora y fecha definidas
- Tras un número definido de apariciones

También puede especificar que las apariciones continúen para siempre.

**V 9.1.0.5** Si una transferencia planificada se ejecuta a la misma hora cada día, utilice el atributo **adjustScheduleTimeForDaylightSaving** en el archivo de propiedades del agente para ajustar la hora a la que se produce la planificación cuando cambian los relojes. Consulte [El archivo MFT agent.properties](#) para obtener más información.

Para crear una nueva transferencia de archivos planificada utilizando la línea de mandatos, utilice los parámetros de planificación (**-tb**, **-ss**, **-oi**, **-of**, **-oc** y **-es**) para el [mandato fteCreateTransfer](#).

Para crear una nueva transferencia de archivos planificada utilizando el asistente **Crear nueva transferencia de archivos gestionada** en IBM MQ Explorer, utilice los pasos siguientes:

### Procedimiento

1. En la vista Navegador, pulse **Transferencia de archivos gestionada**. En la vista Contenido, aparece **Central de transferencias de archivos gestionadas**.
2. En la vista Navegador, aparecen todos los gestores de colas de coordinación. Expanda el nombre del gestor de colas de coordinación en el que está registrado el agente que desea utilizar para la transferencia. Si está conectado actualmente a un gestor de colas de coordinación que no sea el que desea utilizar para la transferencia, pulse el botón derecho del ratón en dicho nombre de gestor de colas de coordinación en la vista Navegador y pulse **Desconectar**. A continuación, pulse el botón derecho del ratón en el nombre del gestor de colas de coordinación que desea utilizar y pulse **Conectar**.

3. Inicie el asistente **Crear nueva transferencia de archivos gestionada** mediante uno de los métodos siguientes:
  - a) Pulse el botón derecho del ratón en el nombre de cualquiera de los nodos siguientes en la vista Navegador: el gestor de colas de coordinación relevante, **Plantillas de transferencia**, **Registro de transferencias** o **Transferencias pendientes**. A continuación, pulse **Nueva transferencia** para iniciar el asistente.
  - b) Pulse **Archivo > Nuevo > Otros > Asistentes de transferencia de archivos gestionada > Asistente de nueva transferencia**
4. Siga las instrucciones de los paneles del asistente. Asegúrese de que ha seleccionado el recuadro de selección **Habilitar transferencia de planificación** y de que escribe los detalles de planificación en la pestaña **Planificar**. Las transferencias de archivos planificadas empiezan al cabo de un minuto de la hora de inicio de planificación, si no existen problemas que puedan afectar a la transferencia. Por ejemplo, podrían surgir problemas con la red o el agente que impidan que se inicie la transferencia planificada. Para cada panel se proporciona ayuda según contexto. Para acceder a la ayuda según contexto en Windows, pulse F1. En Linux, pulse Ctrl+F1 o Shift+F1.

## Resultados

Si desea más información sobre los mensajes implicados en transferencias de archivos planificadas, consulte [Formatos de mensaje de registro de transferencia de archivos planificada](#).

## Cómo trabajar con transferencias de archivos pendientes

Puede ver las transferencias de archivos planificadas que están pendientes en IBM MQ Explorer. La ventana **Transferencias pendientes** muestra todas las transferencias pendientes registradas en el gestor de colas de coordinación al que está conectado actualmente.

### Acerca de esta tarea

Para ver el estado de una transferencia de archivos planificada que aún no se ha iniciado, siga estos pasos:

### Procedimiento

1. Expanda **Transferencia de archivos gestionada** en la vista Navegador. En la vista Contenido, aparece **Central de transferencias de archivos gestionadas**.
2. En la vista Navegador, aparecen todos los gestores de colas de coordinación. Expanda el nombre del gestor de colas de coordinación que ha utilizado para la transferencia planificada. Si desea cambiar el gestor de colas de coordinación al que está conectado, pulse el botón derecho del ratón en el nombre del gestor de colas de coordinación que desea utilizar en la vista Navegador y pulse **Conectar**.
3. Pulse **Transferencias pendientes**. En la vista Contenido, aparece **Transferencias pendientes**.
4. La ventana **Transferencias pendientes** muestra los siguientes detalles sobre las transferencias de archivos gestionadas:
  - a) **Nombre** El número de transferencia de archivos planificada. Este número se asigna automáticamente.
  - b) **Origen** El nombre del agente de origen.
  - c) **Archivo de origen** El nombre del archivo que se transferirá al sistema host.
  - d) **Destino** El nombre del agente de destino.
  - e) **Archivo de destino** El nombre del archivo de destino después de que se haya transferido al sistema de destino.
  - f) **Inicio planificado (huso horario seleccionado)** La fecha y la hora en que se ha planificado que se inicie la transferencia de archivos, expresada en el huso horario seleccionado del administrador. Para cambiar el huso horario visualizado, pulse **Ventana > Preferencias > IBM MQ Explorer >**

**Managed File Transfer** y seleccione un huso horario alternativo en la lista **Huso horario**. Pulse **Aceptar**.

- g) **Repetir cada** Si ha elegido repetir la transferencia planificada, el intervalo especificado durante el cual desea repetir la transferencia, expresado como número.
- h) **Tipo de repetición** Si ha elegido repetir la transferencia planificada, el tipo de intervalo de repetición que ha especificado para la transferencia de archivos. El tipo puede ser uno de los valores siguientes: `minutes`, `hours`, `days`, `weeks`, `months` o `years`.
- i) **Repetir** Si ha elegido repetir la transferencia planificada, los detalles de cuándo desea que se detenga la transferencia de archivos de repetición. Por ejemplo, una fecha o una hora específicas o después de un número específico de apariciones.

## Resultados

Para renovar lo que se visualiza en la ventana **Transferencias pendientes**, pulse el botón **Renovar**  en la barra de herramientas de la vista Contenido.

Para cancelar una transferencia de archivos pendiente, pulse el botón derecho del ratón en la transferencia específica y pulse **Cancelar**. La cancelación completa de una transferencia descarta la solicitud de transferencia de archivos.

## Desencadenamiento de una transferencia de archivos

Puede establecer determinadas condiciones desencadenantes en una transferencia de archivos que se deben cumplir para que se pueda producir la transferencia. Si las condiciones desencadenantes no se cumplen, la transferencia de archivos no se produce y se somete un mensaje de registro opcionalmente para registrar el hecho de que no se produjo la transferencia. A continuación, se descarta la petición de transferencia de archivos. Por ejemplo, puede configurar una transferencia de archivos que se produzca sólo si un archivo nombrado en el sistema donde está ubicado el agente de origen sobrepasa un tamaño especificado, o si un determinado archivo nombrado existe en el sistema donde está ubicado el agente de origen. Puede configurar una transferencia de archivos desencadenada desde IBM MQ Explorer o desde la línea de mandatos.

### Acerca de esta tarea

Puede supervisar un recurso continuamente para que se cumpla una condición desencadenante. Para obtener más información sobre la supervisión de recursos, consulte: [“Supervisión de recursos de MFT” en la página 238](#).

Existen tres diferentes condiciones desencadenantes que se pueden establecer. Las condiciones son las siguientes:

- Si un determinado archivo existe en el mismo sistema que el agente de origen
- Si un determinado archivo no existe en el mismo sistema que el agente de origen
- Si un determinado archivo sobrepasa un determinado tamaño en el sistema donde está ubicado el agente de origen (el tamaño se puede expresar en bytes, KB, MB o GB). Estas unidades de medida utilizan la convención  $2^{10}$ , por ejemplo, 1 KB equivale a 1024 bytes y 1 MB equivale a 1024 KB.

Los tipos de desencadenantes de la lista anterior se pueden combinar de dos maneras:

- Para una sola condición, puede especificar más de un archivo en el sistema donde está ubicado el agente de origen. Esto desencadena la transferencia si cualquiera de los archivos especificados cumple la condición (operador booleano OR).
- Puede especificar varias condiciones. Esto desencadena la transferencia únicamente si se cumplen todas las condiciones (operador booleano AND).

También puede combinar la transferencia desencadenada con una transferencia planificada. Para obtener más información, consulte [Creación de una transferencia de archivos planificada](#). En este caso, las



condiciones desencadenantes se evalúan en el momento en el que va a iniciarse la planificación, o bien para una planificación repetitiva cada vez que va a iniciarse la planificación.

Las transferencias desencadenadas no están soportadas en los agentes de puente de protocolo.

Para crear una transferencia de archivos desencadenada utilizando la línea de mandatos, utilice el parámetro **-tr** en el mandato `fteCreateTransfer`.

Para crear una transferencia de archivos planificada utilizando el asistente **Crear nueva transferencia de archivos gestionada** en IBM MQ Explorer, utilice los pasos siguientes:

## Procedimiento


1. En la vista Navegador, pulse **Transferencia de archivos gestionada**. En la vista Contenido, aparece **Central de transferencias de archivos gestionadas**.
2. En la vista Navegador, aparecen todos los gestores de colas de coordinación. Expanda el nombre del gestor de colas de coordinación que ha utilizado para la transferencia planificada. Si desea cambiar el gestor de colas de coordinación al que está conectado, pulse el botón derecho del ratón en el nombre del gestor de colas de coordinación que desea utilizar en la vista Navegador y pulse **Conectar**.
3. Inicie el asistente **Crear nueva transferencia de archivos gestionada** utilizando cualquiera de los dos métodos siguientes:
  - a) Pulse el botón derecho del ratón en el nombre de cualquiera de los nodos siguientes en la vista Navegador: el gestor de colas de coordinación relevante, **Plantillas de transferencia**, **Registro de transferencias** o **Transferencias pendientes**. A continuación, pulse **Nueva transferencia** para abrir el asistente.
  - b) Pulse **Archivo > Nuevo > Otros > Asistentes de transferencia de archivos gestionada > Asistente de nueva transferencia**
4. Siga las instrucciones de los paneles del asistente. Asegúrese de seleccionar el recuadro de selección **Habilitar transferencia desencadenada** en la pestaña **Desencadenantes** y de completar los campos de dicho separador para configurar el desencadenamiento. Para cada panel se proporciona ayuda según contexto. Para acceder a la ayuda según contexto en Windows, pulse F1. En Linux, pulse **Ctrl+F1** o **Shift+F1**.

## Supervisión de transferencias de archivos que están en curso

Puede supervisar una transferencia de archivos que está en curso utilizando la pestaña **Transferencia de archivos gestionada - Progreso de la transferencia actual** en IBM MQ Explorer. Esta transferencia de archivos se puede iniciar desde IBM MQ Explorer o desde la línea de mandatos. la pestaña también muestra el progreso de las transferencias planificadas en el punto de inicio de las transferencias planificadas.

### Acerca de esta tarea

Si desea utilizar IBM MQ Explorer para supervisar transferencias asociadas con un gestor de cola de coordinación en un sistema remoto, siga las instrucciones del tema [“Configuración de IBM MQ Explorer para supervisar un gestor de colas de coordinación remota”](#) en la página 235.

La información de transferencias de archivos anteriores no se mantiene después de detener y reiniciar IBM MQ Explorer. Al reiniciar el sistema, la información sobre las anteriores transferencias se borrará de la pestaña **Progreso de la transferencia actual**. Puede borrar transferencias completas utilizando **Eliminar transferencias completas**  en cualquier momento en el que tenga abierto IBM MQ Explorer.


## Procedimiento


Después de que haya iniciado una nueva transferencia de archivos utilizando IBM MQ Explorer o la línea de mandatos, podrá supervisar el progreso de la transferencia en la pestaña **Progreso de la transferencia actual**. Se visualiza la siguiente información para cada transferencia en curso:

- a) **Origen**. El nombre del agente que se utiliza para transferir el archivo desde el sistema de origen.

- b) **Destino.** El nombre del agente que se utiliza para recibir el archivo en el sistema de destino.
- c) **Archivo actual.** El nombre del archivo que se está transfiriendo actualmente. La parte del archivo individual que se ha transferido se visualiza en B, KiB, MiB, GiB, o TiB junto con el tamaño total del archivo entre paréntesis. La unidad de medida visualizada depende del tamaño del archivo.  
B es bytes por segundos. KiB/s es kibibytes por segundo, donde 1 kibibyte equivale a 1024 bytes. MiB/s es mebibytes por segundo, donde 1 mebibyte equivale a 1 048 576 bytes. GiB/s es gibibytes por segundo, donde 1 gibibyte equivale a 1 073 741 824 bytes. TiB/s es tebibytes por segundo, donde 1 tebibyte equivale a 1 099 511 627 776 bytes.
- d) **Número de archivos.** Si está transfiriendo más de un archivo, este número representa en qué punto del grupo total de archivos se encuentra la transferencia.
- e) **Progreso.** La barra de progreso muestra el porcentaje completado de la transferencia de archivos actual. .
- f) **Velocidad.** La velocidad a la que se transfiere el archivo en KiB/s (kibibytes por segundo, donde 1 kibibyte equivale a 1024 bytes).
- g) **Iniciada (huso horario seleccionado).** Hora a la que se ha iniciado la transferencia de archivos, presentada en el huso horario seleccionado del administrador. Para cambiar el huso horario visualizado, pulse **Ventana > Preferencias > IBM MQ Explorer > Managed File Transfer** y seleccione un huso horario alternativo en la lista **Huso horario**. Pulse **Aceptar**.  
Si la transferencia entra en un estado de recuperación mientras se transfiere el archivo, la hora de inicio se actualiza para reflejar la hora en que se ha reanudado la transferencia de archivo.

## Resultados

Este separador renueva automáticamente la información de forma periódica, pero para forzar una vista renovada de lo que se visualiza en la pestaña **Progreso de la transferencia actual**, pulse **Renovar**  en la barra de herramientas de la vista Contenido.

Para suprimir transferencias de archivo de la pestaña **Progreso de la transferencia actual**, pulse **Eliminar transferencias completas**  en la barra de herramientas de la vista Contenido. Al pulsar este botón, los detalles de la transferencia de archivos sólo se eliminan de la pestaña; no detiene ni cancela una planificación actual o planificada.

Si desea regresar al separador **Progreso de la transferencia actual** después de cerrarlo, puede visualizar la pestaña pulsando en **Ventana > Mostrar vista > Otros > Otros > Transferencia de archivos gestionada - Progreso de la transferencia actual**. Pulse **Aceptar**.

## Qué hacer a continuación

Además, es posible desarrollar aplicaciones para la supervisión de transferencia de archivos personalizada. Esto puede lograrse mediante la creación de una suscripción al tema administrativo de Managed File Transfer adecuado (ya sea mediante programación o de forma administrativa) y la aplicación de supervisor, puede recibir publicaciones de actividad de transferencia de archivos de Managed File Transfer sobre el tema. Si desea más información sobre el tema de suscripción y el formato del mensaje de publicación, consulte [Ejemplos de mensajes de progreso de transferencia de archivos](#).

### Tareas relacionadas

[“Configuración de IBM MQ Explorer para supervisar un gestor de colas de coordinación remota” en la página 235](#)

Utilice IBM MQ Explorer para supervisar las transferencias de archivos asociadas con un gestor de colas de coordinación que se ejecuta en un sistema remoto. En IBM WebSphere MQ 7.5 o posterior necesita un sistema que sea capaz de ejecutar IBM MQ Explorer. El componente IBM MQ Explorer debe estar instalado para poder conectarse al gestor de colas de coordinación remoto.

[“Visualización del estado de transferencias de archivos en el Registro de transferencias” en la página 235](#)  
Puede ver los detalles de las transferencias de archivos utilizando el **Registro de transferencias** en IBM MQ Explorer. Puede tratarse de transferencias iniciadas desde la línea de mandatos o desde IBM MQ Explorer. También puede personalizar lo que aparece en el **Registro de transferencias**.

## Configuración de IBM MQ Explorer para supervisar un gestor de colas de coordinación remota

Utilice IBM MQ Explorer para supervisar las transferencias de archivos asociadas con un gestor de colas de coordinación que se ejecuta en un sistema remoto. En IBM WebSphere MQ 7.5 o posterior necesita un sistema que sea capaz de ejecutar IBM MQ Explorer. El componente IBM MQ Explorer debe estar instalado para poder conectarse al gestor de colas de coordinación remoto.

### Acerca de esta tarea

Suposiciones: Autorización para conectarse al gestor de colas de coordinación remoto configurando el gestor de colas para permitir conexiones remotas.

Si desea más información sobre cómo configurar esto, consulte [Conexión a un gestor de colas en modalidad cliente con autenticación de canal](#) y [Gestión de autoridades para recursos específicos de MFT](#).

Para supervisar gestores de colas y transferencias de archivos entre agentes en un sistema que no ejecuta Windows o Linux, configure IBM MQ Explorer para conectarse al sistema remoto utilizando los pasos siguientes:

### Procedimiento

1. Inicie el IBM MQ Explorer local.
2. Cuando IBM MQ Explorer se haya cargado, pulse el botón derecho del ratón en la carpeta **Transferencia de archivos gestionada** y seleccione **Nueva configuración**.
3. Continúe por el asistente, seleccionado el gestor de colas de coordinación y mandatos y, a continuación, defina un nombre para la configuración.
4. Pulse **Finalizar** para completar la definición.
5. Cuando haya finalizado la definición, pulse el botón derecho del ratón en la definición y seleccione **Connect** (Conectar).

### Resultados

Ahora inicie IBM MQ Explorer y utilícelo para supervisar la actividad de transferencia para la red de Managed File Transfer asociada con el gestor de colas de coordinación.

#### Tareas relacionadas

[“Supervisión de transferencias de archivos que están en curso” en la página 233](#)

Puede supervisar una transferencia de archivos que está en curso utilizando la pestaña **Transferencia de archivos gestionada - Progreso de la transferencia actual** en IBM MQ Explorer. Esta transferencia de archivos se puede iniciar desde IBM MQ Explorer o desde la línea de mandatos. la pestaña también muestra el progreso de las transferencias planificadas en el punto de inicio de las transferencias planificadas.


[“Visualización del estado de transferencias de archivos en el Registro de transferencias” en la página 235](#)

Puede ver los detalles de las transferencias de archivos utilizando el **Registro de transferencias** en IBM MQ Explorer. Puede tratarse de transferencias iniciadas desde la línea de mandatos o desde IBM MQ Explorer. También puede personalizar lo que aparece en el **Registro de transferencias**.

## Visualización del estado de transferencias de archivos en el Registro de transferencias

Puede ver los detalles de las transferencias de archivos utilizando el **Registro de transferencias** en IBM MQ Explorer. Puede tratarse de transferencias iniciadas desde la línea de mandatos o desde IBM MQ Explorer. También puede personalizar lo que aparece en el **Registro de transferencias**.



## Procedimiento

1. Expanda **Transferencia de archivos gestionada** en la vista Navegador y, a continuación, expanda el nombre del gestor de colas de coordinación para el que desea ver el registro de transferencias.
2. Pulse **Registro de transferencias** en la vista Navegador. En la vista Contenido aparece el **Registro de transferencias**.
3. La ventana **Registro de transferencias** muestra la información detallada siguiente sobre las transferencias de archivos:
  - a) **Origen** El nombre del agente en el sistema en el que se encuentra el archivo de origen.
  - b) **Destino** El nombre del agente en el sistema al que desea transferir el archivo.
  - c) **Estado de terminación** El estado de la transferencia de archivos. El estado puede ser uno de los valores siguientes: "Iniciada", "En proceso", "Satisfactoria", "Parcialmente satisfactoria", "Cancelada" o "Fallida".
  - d) **Propietario** El ID de usuario en el host que sometió la petición de transferencia.
  - e) **Iniciada (huso horario seleccionado)** Fecha y hora a la que el agente de Managed File Transfer ha aceptado la solicitud de transferencia de archivo, presentada en el huso horario seleccionado del administrador. Para cambiar el huso horario visualizado, pulse **Ventana > Preferencias > IBM MQ Explorer > Managed File Transfer** y seleccione un huso horario alternativo en la lista **Huso horario**. Pulse **Aceptar**.
  - f) **Estado registrado (huso horario seleccionado)** (Esta columna no aparece de forma predeterminada. Puede elegir visualizar la columna utilizando la ventana **Configurar columnas de registro de transferencias** ) La fecha y hora en que se registró el estado de terminación, en el huso horario seleccionado por el administrador.
  - g) **Nombre de trabajo** Un identificador especificado por el usuario utilizando el parámetro **-jn** de **fteCreateTransfer** o en un script Ant
  - h) **ID de transferencia** El identificador exclusivo de la transferencia de archivo.
  - i) **Connect: Direct** Se listan los detalles sobre **Número de proceso, Nombre de proceso, Nodo primario, Nodo secundario, Tipo de origen y Tipo de destino**.

## Resultados

**Nota:** El formato interno del Registro de transferencias se cambió en IBM MQ 8.0.0 Fix Pack 1 para el APAR IC99545. Como resultado, si se actualiza un IBM MQ Explorer a la versión 8.0.0.1 o posterior y, a continuación, se restaura a la versión 8.0.0.0, no se visualizará ningún XML de auditoría para las transferencias que han tenido lugar mientras IBM MQ Explorer estaba en la versión 8.0.0.1. El panel XML en la ventana **Propiedades** para estas transferencias contendrá un recuadro de texto vacío.

Para ver más detalles sobre una transferencia completada, expanda la transferencia que le interese pulsando el signo más (+). A continuación, puede ver todos los nombres de archivo de origen y destino incluidos en dicha transferencia. No obstante, si la transferencia está actualmente en proceso y consta de muchos archivos, sólo puede ver los archivos que ya se han transferido hasta el momento.

Para renovar lo que se visualiza en el **Registro de transferencias**, pulse el botón **Renovar**  en la barra de herramientas de la vista Contenido. La información de transferencia de archivos en el Registro de transferencias permanece en el registro después de que detenga y reinicie IBM MQ Explorer. Si desea suprimir todas las transferencias de archivo completadas del registro, pulse **Eliminar transferencias completadas**  en la barra de herramientas de la vista Contenido.

Para suprimir una transferencia de archivos individual completada del registro, pulse el botón derecho del ratón en la transferencia y pulse **Suprimir**. Si suprime una transferencia, no se detiene o cancela una transferencia que está en proceso o que se ha planificado; se suprimen solamente los datos históricos almacenados.

Para copiar el identificador exclusivo de una transferencia en el portapapeles, pulse el botón derecho del ratón en esa transferencia y pulse **Copiar ID**.

Los metadatos y el XML de la auditoría completa de la transferencia están disponibles en el menú emergente, en la acción **Propiedades**.

### Tareas relacionadas

[“Supervisión de transferencias de archivos que están en curso” en la página 233](#)

Puede supervisar una transferencia de archivos que está en curso utilizando la pestaña **Transferencia de archivos gestionada - Progreso de la transferencia actual** en IBM MQ Explorer. Esta transferencia de archivos se puede iniciar desde IBM MQ Explorer o desde la línea de mandatos. la pestaña también muestra el progreso de las transferencias planificadas en el punto de inicio de las transferencias planificadas.

[“Configuración del registro de transferencias” en la página 237](#)

Puede configurar qué información desea visualizar y cómo se visualiza en el **Registro de transferencias** es IBM MQ Explorer.

[“Establecimiento de un tiempo de espera para la recuperación de transferencias estancadas” en la página 315](#)

Puede establecer un tiempo de espera de recuperación de transferencia para las transferencias de archivos estancadas que se aplique a todas las transferencias para un agente de origen. También puede establecer un tiempo de espera de recuperación de transferencia para una transferencia individual. Si establece una cantidad de tiempo específica, en segundos, durante la cual un agente de origen sigue intentando recuperar una transferencia de archivos estancada y la transferencia no se realiza correctamente cuando el agente alcanza el tiempo de espera, la transferencia falla.


## Configuración del registro de transferencias

Puede configurar qué información desea visualizar y cómo se visualiza en el **Registro de transferencias** es IBM MQ Explorer.


### Acerca de esta tarea

Para reorganizar el orden de las columnas en **Registro de transferencias**, pulse el título de la columna que desea mover y arrastre la columna hasta la nueva posición. El nuevo orden de columnas sólo se mantendrá hasta la próxima vez que detenga y reinicie IBM MQ Explorer.

Para filtrar entradas en el **Registro de transferencias**, especifique una serie en el campo **Filtrar las entradas de registro visualizadas**. Para restaurar todas las entradas en el registro, suprima la serie especificada del campo. Se puede utilizar cualquier expresión regular Java válida en este campo. Si desea más información, consulte [Expresiones regulares utilizadas por MFT](#).

Para personalizar qué columnas se visualizan en el Registro de transferencias, utilice **Configurar columnas del registro de transferencias** . Utilice los pasos siguientes para iniciar y utilizar la ventana **Configurar columnas del registro de transferencias**.

### Procedimiento

1. Asegúrese de que el **Registro de transferencias** está abierto en la vista Contenido. Pulse **Configurar columnas de registro de transferencias**  en la barra de herramientas de la vista Contenido. Se abre la ventana **Configurar columnas del registro de transferencias**.
2. Para personalizar la vista del **Registro de transferencias**, seleccione o deseleccione los recuadros de selección individuales de las columnas que desea mostrar u ocultar. Puede pulsar **Seleccionar todo** y luego **Aceptar** para seleccionar todos los recuadros de selección o **Deseleccionar todo** y luego **Aceptar** para borrar la marca de selección de todos los recuadros.

### Tareas relacionadas

[“Supervisión de transferencias de archivos que están en curso” en la página 233](#)

Puede supervisar una transferencia de archivos que está en curso utilizando la pestaña **Transferencia de archivos gestionada - Progreso de la transferencia actual** en IBM MQ Explorer. Esta transferencia de archivos se puede iniciar desde IBM MQ Explorer o desde la línea de mandatos. la pestaña también

muestra el progreso de las transferencias planificadas en el punto de inicio de las transferencias planificadas.

“[Visualización del estado de transferencias de archivos en el Registro de transferencias](#)” en la página 235 Puede ver los detalles de las transferencias de archivos utilizando el **Registro de transferencias** en IBM MQ Explorer. Puede tratarse de transferencias iniciadas desde la línea de mandatos o desde IBM MQ Explorer. También puede personalizar lo que aparece en el **Registro de transferencias**.

## Supervisión de recursos de MFT

Puede supervisar recursos de Managed File Transfer; por ejemplo, una cola o un directorio. Cuando se cumple una condición en este recurso, el supervisor de recursos inicia una tarea, por ejemplo una transferencia de archivos. Puede crear un supervisor de recursos utilizando el mandato **fteCreateMonitor** o la vista **Supervisores** en el plug-in de Managed File Transfer para IBM MQ Explorer.

### Acerca de esta tarea

La supervisión de recursos de Managed File Transfer utiliza la siguiente terminología:

#### **Supervisor de recursos**

Un supervisor de recursos es un proceso que sondea un recurso (por ejemplo, un directorio o una cola) a intervalos regulares predefinidos para ver si el contenido del recurso ha cambiado. En caso de que haya cambiado, el contenido se compara con el conjunto de condiciones de este supervisor. Si se encuentra alguna coincidencia, se iniciará la tarea de este supervisor.

#### **Recurso**

Recurso del sistema que el supervisor de recursos examina en cada intervalo de sondeo para compararlo con las condiciones de desencadenante. Colas, directorios o estructuras de directorios anidados pueden ser el recurso supervisado.

#### **Condición y condición de desencadenante**

Una condición es una expresión que se evalúa (normalmente, respecto al contenido del recurso supervisado). Si la expresión se evalúa como true (verdadera), la condición contribuye a la condición desencadenante global.

La condición de desencadenante es la condición global, que se cumple cuando se cumplen todas las condiciones. Cuando se cumple la condición desencadenante, la tarea puede proseguir.

#### **Tarea**

Una tarea es la operación que se inicia cuando se cumple la condición o el conjunto de condiciones de desencadenante. Las tareas soportadas son la transferencia de archivos y la llamada de mandatos.

#### **Archivo desencadenante**

Un archivo desencadenante es un archivo que se coloca en un directorio supervisado para indicar que se puede iniciar una tarea (normalmente, una transferencia). Por ejemplo, podría indicar que todos los archivos que se van procesar han llegado a una ubicación conocida y se pueden transferir o pueden ser objeto de otras operaciones. El nombre del archivo desencadenante se puede utilizar para especificar los archivos que se van a transferir, mediante la sustitución de variables. Para obtener más información, consulte “[Personalización de las tareas del supervisor de recursos de MFT con sustitución de variables](#)” en la página 249.

El archivo desencadenante también se conoce como archivo preparado o archivo en formato go. Sin embargo, en esta documentación se suele denominar archivo desencadenante.

La supervisión de recursos no está soportada en agentes de puente de protocolo o agentes de puente de Connect:Direct.

## Conceptos de supervisión de recursos de MFT

Una visión general de los conceptos clave de la característica de supervisión de recursos de Managed File Transfer.



## Supervisores de recursos

Puede crear un supervisor de recursos utilizando el mandato **fteCreateMonitor**, que crea e inicia un nuevo supervisor de recursos desde la línea de mandatos. El supervisor de recursos está asociado a un agente de Managed File Transfer y sólo se activa cuando se inicia dicho agente. Cuando se detiene el agente de supervisión, también lo hace el supervisor. Si el agente ya se ha iniciado cuando se crea el supervisor de recursos, el supervisor de recursos se inicia de forma inmediata. El agente de supervisión también debe ser el agente de origen de la tarea iniciada por el supervisor de recursos.

Los nombres de supervisor de recursos deben ser exclusivos dentro del agente. El nombre del supervisor de recursos debe tener un mínimo de un carácter de longitud y no debe contener caracteres de asterisco (\*), porcentaje (%) o signo de interrogación (?). Las mayúsculas y minúsculas que se incluyen en un nombre de supervisor de recursos se omiten y el nombre de supervisor de recursos se convierte a mayúsculas. Si intenta crear un supervisor de recursos con un nombre que ya existe, la solicitud se ignora y el intento se registra en el tema de registro del supervisor de recursos.

**Nota:** **V 9.1.0** No puede crear un supervisor de recursos con una definición de tarea que contenga transferencias planificadas.

No existe ninguna restricción en cuanto al número de supervisores de recursos que se pueden crear en un agente y todos ellos se ejecutan con la misma prioridad. Tenga en cuenta las implicaciones del solapamiento de recursos supervisados, de las condiciones desencadenantes en conflicto y de la frecuencia con la que se sondan los recursos.

Los supervisores de recursos solapados pueden causar:

- Posible contención en la ubicación/elementos de origen.
- Posibles solicitudes de transferencia duplicadas para los mismos elementos de origen.
- Anomalías o errores inesperados para las transferencias debido a conflictos de elementos de origen.

Si varios supervisores exploran la misma ubicación y se pueden desencadenar en los mismos elementos, puede potencialmente terminar con el problema de dos supervisores diferentes que envían solicitudes de transferencia gestionadas para el mismo elemento.

Los supervisores de recursos consultan el contenido de los recursos después de cada periodo de intervalo de sondeo. El contenido del recurso se compara con las condiciones desencadenantes y, si se cumplen estas condiciones, se llama a la tarea asociada al supervisor de recursos.

La tarea se inicia asíncronamente. Si se encuentra una coincidencia de condición y se inicia la tarea, el supervisor de recursos continúa sondeando para realizar más cambios en el contenido del recurso. Así, por ejemplo, si se ha producido una coincidencia porque un archivo denominado `reports.go` ha llegado a un directorio supervisado, la tarea se iniciaría una vez. En el próximo intervalo de sondeo, aunque el archivo siga existiendo, la tarea no se vuelve a iniciar. No obstante, si el archivo se suprime y luego se vuelve a colocar en el directorio, o si el archivo se actualiza (de forma que se cambia el atributo de fecha de última modificación), la siguiente comprobación de condición desencadenante hace que se vuelva a invocar la tarea.

En versiones anteriores a IBM MQ 9.1.5, si un supervisor de recursos realiza un sondeo que lleva más tiempo que el intervalo de sondeo, esto significa que el sondeo siguiente se inicia tan pronto como finaliza el actual sin espacio entre ellos, lo que podría tener afectar a la agilidad con que los supervisores de recursos envían trabajo a un agente. Esto podría provocar problemas de rendimiento si los elementos que se encuentran durante el primer sondeo siguen allí cuando se produce el segundo.

**V 9.1.5** A partir de IBM MQ 9.1.5, el supervisor de recursos utiliza `ScheduledExecutorService` e inicia el siguiente sondeo solo después de la finalización del sondeo anterior más el tiempo de intervalo de sondeo configurado. Esto significa que siempre habrá una brecha entre los intervalos de sondeo, en lugar de tener otro sondeo que empiece inmediatamente después del sondeo anterior, si el tiempo de sondeo era más largo que el intervalo de sondeo.

**V 9.1.3** A partir de IBM MQ 9.1.3, si un archivo no se ha podido transferir, puede borrar el historial del supervisor de recursos, lo que permite que se someta otra solicitud de transferencia sin que sea

necesario suprimir el archivo y colocarlo en el directorio de nuevo, o actualizar el archivo para cambiar el atributo de última fecha de modificación. El borrado del historial es útil, por ejemplo, en situaciones en las que es necesario transferir el archivo, pero no es posible modificarlo. Para obtener más información, consulte [“Borrado del historial del supervisor de recursos”](#) en la página 264.

## Recursos

Los supervisores de recursos de Managed File Transfer pueden sondear el contenido de los dos tipos de recursos siguientes:

### Directorios o estructuras de directorios anidados

Un escenario habitual es supervisar un directorio para ver si hay un archivo desencadenante. Es posible que una aplicación externa procese varios archivos y los coloque en un directorio de origen conocido. Cuando la aplicación ha terminado de procesarse, indica que los archivos están listos para su transferencia o pueden ser objeto de otras operaciones, colocando un archivo desencadenante en una ubicación supervisada. El archivo desencadenante puede ser detectado por un supervisor de recursos de Managed File Transfer y se inicia la transferencia de esos archivos del directorio de origen a otro Managed File Transfer Agent.

De forma predeterminada, se supervisa el directorio especificado. Para examinar también los subdirectorios, establezca el nivel de recurrencia en el mandato **fteCreateTransfer**.

Los siguientes son dos ejemplos de supervisión de un directorio:

- Supervise un archivo desencadenante (por ejemplo, `trigger.file`) y luego transfiera un comodín (por ejemplo, `*.zip`).
- Supervise `*.zip` y luego transfiera `${FilePath}` (por ejemplo, el archivo que ha desencadenado la transferencia). Para obtener más información sobre la sustitución de variables, consulte [“Personalización de las tareas del supervisor de recursos de MFT con sustitución de variables”](#) en la página 249.

**Nota:** No cree un supervisor que supervise `*.zip` y, a continuación, transfiera `*.zip`. El supervisor intenta iniciar una transferencia de `*.zip` para cada archivo `.zip` del sistema. Es decir, el supervisor genera \* número de transferencias para `*.zip`.

Para obtener un ejemplo de la creación de un supervisor de recursos para supervisar un directorio, consulte [“Supervisar un directorio y utilizar la sustitución de variables”](#) en la página 247.

### Colas de IBM MQ

Un ejemplo de supervisión de un cola es que una aplicación externa puede estar generando mensajes y colocándolos en una cola conocida con el mismo ID de grupo. Cuando la aplicación ha terminado de poner mensajes en la cola, esto indica que el grupo está completo. El grupo completo de mensajes puede ser detectado por un supervisor de recursos de Managed File Transfer y se inicia la transferencia de un grupo de mensajes de la cola de origen a un archivo. Para obtener un ejemplo de la creación de un supervisor de recursos para supervisar una cola, consulte [“Ejemplo: configuración de un recurso MFT”](#) en la página 248.

**Nota:** Puede especificar un solo supervisor por cola. Si especifica más de un supervisor para sondear una cola de IBM MQ, se produce un comportamiento impredecible.

La supervisión de conjuntos de datos no está soportada.

## Condiciones y condiciones de desencadenante

La condición se cumple cuando el recurso contiene un valor que coincide con alguna serie o algún patrón. Las condiciones pueden ser una de las siguientes:

- Coincidencia de nombres de archivo (patrón)
- Ninguna coincidencia en el nombre de archivo (patrón)
- Tamaño de archivo
- Coincidencia si el tamaño de archivo sigue siendo el mismo para un número de sondeos



La coincidencia de nombres de archivos se puede expresar como:

- Coincidencia exacta de series
- Coincidencia de comodines simples como se describe en [Utilización de caracteres comodín con MFT](#)
- Coincidencia de expresiones regulares

Los nombres de archivo también se pueden excluir de la comparación de nombres de archivo utilizando un comodín o una expresión regular Java que identifique los nombres de archivo que nunca coinciden.

Cuando se detecta un archivo coincidente, se conserva la indicación de fecha y hora de última modificación. Si en sondeos posteriores se detecta que el archivo ha cambiado, la condición desencadenante se cumple de nuevo y la tarea se inicia. Si la condición consiste en detectar cuándo un archivo no existe, si ningún archivo del directorio supervisado coincide con el patrón de nombres de archivos, la tarea se inicia. Si luego se añade un archivo al directorio que sí coincide con el patrón de nombre de archivo, la tarea sólo se inicia si luego el archivo se suprime.

## Tareas

Managed File Transfer da soporte a los dos tipos de tareas siguientes que se pueden configurar para que las inicien los supervisores de recursos:

### Tareas de transferencia de archivos

Las tareas de transferencia de archivos se definen de la misma forma que cualquier otra transferencia de archivos. Una forma útil de generar el XML de tarea que necesita un supervisor es ejecutar el mandato `fteCreateTransfer` con el parámetro `-gt`. Este mandato genera una definición de tarea como un documento XML, incluyendo la especificación de la transferencia. A continuación, pasa el nombre del documento XML de tarea como valor del parámetro `-mt` en el mandato `fteCreateMonitor`. Cuando el mandato `fteCreateMonitor` se ejecuta, lee el documento XML de tarea. Una vez ejecutado el mandato `fteCreateMonitor`, los cambios realizados en el archivo XML de tarea no son utilizados por el supervisor.

Cuando se utiliza una tarea de transferencia de archivos, puede seleccionar cuántas condiciones desencadenantes se procesan por lotes en una tarea. El valor predeterminado es que una condición desencadenante inicie una tarea. Puede ejecutar el mandato `fteCreateMonitor` con la opción `-bs` para seleccionar el número de condiciones de desencadenante que se agrupan en una tarea.

### Tareas de mandato

Las tareas de mandato pueden ejecutar scripts Ant, llamar a programas ejecutables o ejecutar trabajos JCL. Para obtener más información, consulte [“Configuración de tareas de supervisión de MFT para iniciar mandatos y scripts”](#) en la página 243.

## Archivos desencadenantes

Puede utilizar el contenido de un archivo desencadenante en un supervisor de recursos para definir un conjunto de archivos que deben transferirse en una única solicitud de transferencia. Cada vez que se detecte un archivo desencadenante coincidente, su contenido se analizará en busca de vías de acceso de archivo de origen y, opcionalmente, vías de acceso de archivo de destino. Estas vías de acceso de archivo se utilizarán para definir elementos del archivo XML de transferencia de tarea que especifique, que se someterá como una única solicitud de transferencia al agente. La definición del supervisor de recursos determina si el contenido del desencadenante está habilitado.

El formato de cada archivo desencadenante es una única vía de acceso de archivo que debe transferirse en cada línea de texto. El formato predeterminado de la línea es una única vía de acceso de archivo de origen, o una vía de acceso de origen y destino separada por una coma.

Para obtener más información y ejemplos, consulte [“Utilización de un archivo desencadenante”](#) en la página 257.

## Copia de seguridad y restauración de supervisores de recursos

Puede hacer copia de seguridad de los supervisores de recursos ya definidos para poder volver a utilizarlos en el futuro. Hay varias opciones que puede utilizar como se indica a continuación:

- Utilice el mandato **fteCreateMonitor** con el parámetro **-ox** para exportar una configuración de supervisor de recursos a un archivo XML y con el parámetro **-ix** para restaurar un supervisor de recursos importando la configuración de supervisor de recursos desde un archivo XML.
- Utilice el mandato **fteListMonitors** con **-ox** para exportar la definición de un único supervisor de recursos a un archivo XML.
- **V9.1.0** Utilice el mandato **fteListMonitors** con **-od** para exportar varias definiciones de supervisor de recursos a un directorio especificado. Cada definición de supervisor de recursos se guarda en el archivo XML aparte. También puede utilizar la opción **-od** para exportar una única definición de supervisor de recursos a un directorio especificado.

Para obtener más información, consulte [“Copia de seguridad y restauración de supervisores de recursos de MFT”](#) en la página 262.

## Registro del supervisor de recursos

**V9.1.0**

A partir de IBM MQ 9.1.0, Managed File Transfer incluye el registro del supervisor de recursos. Para obtener más información, consulte [“Registro de supervisores de recursos de MFT”](#) en la página 260.

### Conceptos relacionados

[“Personalización de las tareas del supervisor de recursos de MFT con sustitución de variables”](#) en la página 249

Cuando se cumplen las condiciones desencadenantes de un supervisor de recursos activo, se llama a la tarea definida. Además de llamar a la tarea de transferencia o mandato con el mismo agente de destino o el mismo nombre de archivo de destino cada vez, también puede modificar la definición de tarea durante la ejecución. Para ello, inserte los nombres de variables en el XML de definición de tarea. Cuando el supervisor determina que se cumplen las condiciones de desencadenante y que la definición de tarea contiene nombres de variable, sustituye los nombres de variable por los valores de variable y, a continuación, llama a la tarea.

### Tareas relacionadas

[“Configuración de tareas de supervisión de MFT para iniciar mandatos y scripts”](#) en la página 243

Los supervisores de recursos no se limitan a realizar transferencias de archivos y la tarea asociada. También puede configurar el supervisor para invocar otros mandatos desde el agente de supervisión, incluidos los programas ejecutables, los scripts Ant o los trabajos JCL. Para llamar a los mandatos, edite el formato XML de definición de tarea del supervisor para incluir uno o varios elementos de mandatos con los parámetros de llamada de mandatos correspondientes, como por ejemplo, argumentos y propiedades.

[“Ejemplo: configuración de un recurso MFT”](#) en la página 248

Puede especificar una cola de IBM MQ como el recurso que va a supervisar un supervisor de recursos utilizando el parámetro **-mq** con el mandato **fteCreateMonitor**.

[“Supervisión de una cola y utilización de sustitución de variables”](#) en la página 254

Puede supervisar una cola y transferir mensajes de la cola supervisada a un archivo utilizando el mandato **fteCreateMonitor**. El valor de cualquier propiedad de mensaje de IBM MQ en el primer mensaje que se va a leer de la cola supervisada se puede sustituir en la definición XML de la tarea y se utiliza para definir el comportamiento de la transferencia.

### Referencia relacionada

**fteCreateMonitor**: crear un supervisor de recursos de MFT

**fteListMonitors**: listar supervisores de recursos de MFT

**FteDeleteMonitor**: suprimir un supervisor de recursos de MFT

## Configuración de tareas de supervisión de MFT para iniciar mandatos y scripts

Los supervisores de recursos no se limitan a realizar transferencias de archivos y la tarea asociada. También puede configurar el supervisor para invocar otros mandatos desde el agente de supervisión, incluidos los programas ejecutables, los scripts Ant o los trabajos JCL. Para llamar a los mandatos, edite el formato XML de definición de tarea del supervisor para incluir uno o varios elementos de mandatos con los parámetros de llamada de mandatos correspondientes, como por ejemplo, argumentos y propiedades.

### Acerca de esta tarea

La vía de acceso de archivos al programa ejecutable, script Ant o trabajo JCL al que desea que el agente de supervisión llame, debe estar incluido en `commandPath` del agente de supervisión. Si desea más información sobre la propiedad de vía de acceso de mandato, consulte [Propiedad `commandPath` de MFT](#).

Puede crear el documento XML de definición de tarea en una de las formas siguientes:

- Cree el documento XML de definición de tarea manualmente de acuerdo con el esquema de `FileTransfer.xsd`. Para obtener más información, consulte [“Cree el XML de definición de tarea manualmente según el esquema”](#) en la página 243.
- Edite el documento XML generado por el parámetro `fteCreateTransfer -gt` como base de la definición de tarea. Para obtener más información, consulte [“Creación de un documento de definición de tarea modificando un documento generado”](#) en la página 245.

Si desea una tarea de transferencia o una tarea de mandato, la definición de tarea debe empezar por un elemento raíz `<request>`. El elemento hijo de `<request>` debe ser `<managedTransfer>` o `<managedCall>`. Normalmente, se selecciona `<managedCall>` cuando hay un solo mandato o script para ejecutar y `<managedTransfer>` si desea que la tarea incluya una transferencia de archivos y opcionalmente, hasta cuatro llamadas de mandatos.

### Cree el XML de definición de tarea manualmente según el esquema

#### Acerca de esta tarea

Puede crear manualmente un archivo XML de definición de tarea de acuerdo con el esquema `FileTransfer.xsd`. Este esquema se puede encontrar en `MQ_INSTALLATION_PATH/mqft/samples/schema`. Si desea más información sobre este esquema, consulte [Formato de mensaje de solicitud de transferencia de archivos](#).

#### Ejemplo

El ejemplo siguiente muestra un documento XML de definición de tarea de ejemplo guardado como `cleanuptask.xml`, que utiliza el elemento `<managedCall>` para llamar a un script Ant denominado `RunCleanup.xml`. El script `RunCleanup.xml` Ant debe estar ubicado en `commandPath` del agente de supervisión.

```
<?xml version="1.0" encoding="UTF-8"?>
<request version="4.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedCall>
    <originator>
      <hostname>hostname</hostname>
      <userID>userID</userID>
      <mqmdUserID>mqmdUserID</mqmdUserID>
    </originator>
    <agent QMgr="QM1" agent="AGENT1"/>
    <reply QMGR="QM1">reply</reply>
    <transferSet priority="1">
      <metaDataSet>
        <metaData key="name1">value1</metaData>
      </metaDataSet>
      <call>
        <command name="RunCleanup.xml" type="antscript" retryCount="2"
          retryWait="30" successRC="0">
```

```

    <target>check_exists</target>
    <target>copy_to_archive</target>
    <target>rename_temps</target>
    <target>delete_files</target>
    <property name="trigger.filename" value="{FileName}"/>
    <property name="trigger.path" value="{FilePath}"/>
  </command>
</call>
</transferSet>
<job>
  <name>JOBCLEAN1</name>
</job>
</managedCall>
</request>

```

El elemento `<agent>` especifica el Managed File Transfer Agent que se configura con el script Ant con nombre en su `commandPath`.

La estructura `<call><command>` . . . define el ejecutable o script que desea ejecutar. El mandato toma un atributo `type` opcional que puede tener uno de los valores siguientes:

#### **antscript**

Ejecute un script Ant en una JVM por separado.

#### **executable**

Invoque un programa ejecutable.

#### **jcl**

Invoque un trabajo JCL.

Si omite el atributo `type`, se utiliza el valor predeterminado `executable`.

El atributo `name` especifica el nombre del script Ant, del ejecutable o del trabajo JCL que desea ejecutar, sin ninguna información de vía de acceso. El agente busca el script o programa en las ubicaciones especificadas por la propiedad `commandPath` en el archivo `agent.properties` del agente.

El atributo `retrycount` especifica el número de veces que se intenta volver a llamar al programa si el programa no devuelve un código de retorno correcto. El valor asignado a este atributo no debe ser negativo. Si no especifica el atributo `retrycount`, se utiliza un valor predeterminado de cero.

El atributo `retrywait` especifica el tiempo de espera, en segundos, antes de volver a intentar la invocación del programa. El valor asignado a este atributo no debe ser negativo. Si no especifica el atributo `retrywait`, se utiliza un valor predeterminado de cero.

El atributo `successrc` es una expresión utilizada para determinar cuándo se ejecuta correctamente la invocación del programa. El código de retorno del proceso para el mandato se evalúa mediante esta expresión. El valor puede constar de una o varias expresiones combinadas con un carácter de barra vertical (|) que indica el booleano OR o un carácter ampersand (&) que indica el booleano AND. Cada expresión puede ser uno de los tipos siguientes de expresión:

- Un número que indica una prueba de igualdad entre el código de retorno del proceso y el número.
- Un número que tiene como prefijo un carácter mayor que (>) para indicar una prueba mayor que entre el número y el código de retorno del proceso.
- Un número que tiene como prefijo un carácter menor que (<) para indicar una prueba menor que entre el número y el código de retorno del proceso.
- Un número precedido de un carácter de signo de exclamación (!) para indicar una prueba de no igualdad entre el número y el código de retorno del proceso. Por ejemplo: `>2&<7&!5|0|14` significa que los códigos de retorno siguientes son satisfactorios: 0, 3, 4, 6, 14. Los demás códigos de retorno se interpretan como no satisfactorios.

Si no especifica el atributo `successrc`, se utiliza un valor predeterminado de cero. Esto significa que se interpreta que el mandato se ha ejecutado satisfactoriamente si, y solamente si, dicho mandato devuelve un código de cero.

Para un script Ant, lo normal es especificar los elementos `<target>` y `<property>`. Los valores del elemento `<target>` deben coincidir con los nombres de destino en el script Ant.

Para programas ejecutables, puede especificar elementos <argument>. Los elementos de argumentos anidados especifican argumentos para transferir al programa que se invoca como parte de la invocación de programa. Los argumentos del programa se crean a partir de los valores especificados por los elementos de argumento en el orden en que se encuentran los elementos de argumento. Puede especificar cero o varios elementos de argumento de una invocación de programa.

El administrador define e inicia el supervisor como normal utilizando el documento XML de definición de tarea que incluye el elemento <managedCall>. Por ejemplo:

```
fteCreateMonitor -ma AGENT1 -mm QM1 -md /monitored -mn MONITOR01 -mt  
/tasks/cleanuptask.xml -pi 30 -pu seconds -tr match,*go
```

La vía de acceso al documento XML de definición de transferencia debe estar en el sistema de archivos local desde el que se ejecuta el mandato **fteCreateMonitor** (en este ejemplo, /tasks/cleanuptask.xml). El documento cleanuptask.xml se utiliza sólo para crear el supervisor de recursos. Las tareas que las referencias de documento cleanuptask.xml (scripts Ant o trabajos JCL) deben estar en la vía de acceso del mandato del agente de supervisión. Cuando se cumple la condición desencadenante del supervisor, las variables del documento XML de definición de tarea se sustituyen por los valores reales del supervisor. Así, por ejemplo, \${FilePath} se sustituye en el mensaje de solicitud enviado al agente por /monitored/cleanup.go. El mensaje de solicitud se pone en la cola de mandatos de agente. El procesador de mandatos detecta que la solicitud es para una llamada de programa e inicia el programa especificado. Si se llama a un mandato de tipo antscript, se inicia una nueva JVM y la tarea Ant se ejecuta bajo la nueva JVM. Para obtener más información sobre la utilización de sustitución de variables, consulte [Personalización de tareas con la sustitución de variables](#).

### Conceptos relacionados

[“Personalización de las tareas del supervisor de recursos de MFT con sustitución de variables” en la página 249](#)

Cuando se cumplen las condiciones desencadenantes de un supervisor de recursos activo, se llama a la tarea definida. Además de llamar a la tarea de transferencia o mandato con el mismo agente de destino o el mismo nombre de archivo de destino cada vez, también puede modificar la definición de tarea durante la ejecución. Para ello, inserte los nombres de variables en el XML de definición de tarea. Cuando el supervisor determina que se cumplen las condiciones de desencadenante y que la definición de tarea contiene nombres de variable, sustituye los nombres de variable por los valores de variable y, a continuación, llama a la tarea.

### Referencia relacionada

[Formato de mensaje de solicitud de transferencia de archivos](#)

[Propiedad commandPath de MFT](#)

## ***Creación de un documento de definición de tarea modificando un documento generado***

### Acerca de esta tarea

Puede crear el documento de definición de tarea del supervisor modificando el documento XML generado por la opción **-gt** de **fteCreateTransfer**. El documento generado tiene un elemento <request> seguido del elemento <managedTransfer>. Para convertir esta definición de tarea en una estructura de <managedCall> válida, siga estos pasos:

### Procedimiento

1. Sustituya las etiquetas de inicio y finalización de <managedTransfer> por etiquetas <managedCall>.
2. Elimine los nodos hijo y elemento <schedule>.
3. Sustituya las etiquetas de inicio y finalización de <sourceAgent> por <agent> para que coincidan con los detalles de configuración del agente de supervisión.
4. Elimine los elementos <destinationAgent> y <trigger>.
5. Elimine los elementos <item>.

6. Inserte una nueva estructura `<call>...</call>` dentro del elemento `<transferSet>`. Esta estructura contiene la definición de mandato tal como se muestra en el ejemplo siguiente:

```
<call>
  <command name="RunCleanup.xml" type="antscript" retryCount="2"
    retryWait="30" successRC="0">
    <target>check_exists</target>
    <target>copy_to_archive</target>
    <target>rename_temps</target>
    <target>delete_files</target>
    <property name="trigger.filename" value="{FileName}"/>
    <property name="trigger.path" value="{FilePath}"/>
  </command>
</call>
```

### Ejemplo

También puede retener el elemento `<managedTransfer>` incluidos todos los detalles de transferencia de archivos e insertar hasta cuatro llamadas de mandatos. En este caso, inserta cualquier selección de los siguientes elementos de llamada entre los elementos `<metaDataSet>` y `<item>`:

#### preSourceCall

Llame a un programa en el agente de origen antes de iniciar la transferencia.

#### postSourceCall

Llame a un programa en el agente de origen después de completar la transferencia.

#### preDestinationCall

Llame a un programa en el agente de destino antes de iniciar la transferencia.

#### postDestinationCall

Llame a un programa en el agente de destino después de completar la transferencia.

Cada uno de estos elementos toma la estructura del elemento `<command>` tal como se describe en el ejemplo anterior. El esquema `FileTransfer.xsd` define los tipos utilizados por los diversos elementos de llamada.

En el ejemplo siguiente se muestra `preSourceCall`, `postSourceCall`, `preDestinationCall` y `postDestinationCall` en un documento de definición de tarea:

```
:
<transferSet priority="1">
  <metaDataSet>
    <metaData key="key1">value1</metaData>
  </metaDataSet>
  <preSourceCall>
    <command name="send.exe" retryCount="0" retryWait="0" successRC="0"
      type="executable">
      <argument>report1.pdf</argument>
      <argument>>true</argument>
    </command>
  </preSourceCall>
  <postSourceCall>
    <command name="//DO_IT.JCL" retryCount="0" retryWait="0" successRC="0"
      type="jcl">
      <argument>argument</argument>
    </command>
  </postSourceCall>
  <preDestinationCall>
    <command name="ant_script.xml" retryCount="0" retryWait="0" successRC="0"
      type="antscript">
      <target>step1</target>
      <property name="name" value="value"/>
    </command>
  </preDestinationCall>
  <postDestinationCall>
    <command name="runit.cmd" retryCount="0" retryWait="0" successRC="0" />
  </postDestinationCall>
  <item checksumMethod="none" mode="binary">
:
```

Puede combinar diferentes tipos de mandato en la transferencia. Los elementos de argumento, destino y propiedad son opcionales.

## Supervisar un directorio y utilizar la sustitución de variables

Puede supervisar un directorio utilizando el mandato **fteCreateMonitor**. El valor de una variable de sustitución se puede sustituir en la definición del XML de tarea y se puede utilizar para definir el comportamiento de la transferencia.

### Acerca de esta tarea

En este ejemplo, el agente de origen se denomina AGENT\_HOP. El directorio que AGENT\_HOP supervisa se llama /test/monitored. El agente sondea el directorio cada 5 minutos.

Después de escribir un archivo .zip en el directorio, la aplicación que escribe el archivo en el directorio escribe un archivo desencadenante en el mismo directorio. El nombre del archivo desencadenante es el mismo que el nombre del archivo .zip, pero tiene una extensión de archivo diferente. Por ejemplo, después de escribir el archivo file1.zip en el directorio, el archivo file1.go se escribe en el directorio. El supervisor de recursos supervisa el directorio de los archivos que coinciden con el patrón \*.go y, a continuación, utiliza la sustitución de variable para solicitar una transferencia del archivo .zip asociado.

### Procedimiento

1. Cree el XML de tarea que define la tarea que el supervisor lleva a cabo cuando se desencadena.

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>blue.example.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_HOP" QMgr="QM_HOP" />
    <destinationAgent agent="AGENT_SKIP" QMgr="QM_SKIP" />
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <file>/test/monitored/_${fileName}{token=1}{separator=.}.zip</file>
        </source>
        <destination type="file" exist="overwrite">
          <file>/out/_${fileName}{token=1}{separator=.}.zip</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Las variables que se sustituyen por los valores asociados al archivo desencadenante están resaltadas en **negrita**. Este XML de tarea se guarda en el archivo /home/USER1/task.xml

2. Cree un supervisor de recursos para supervisar el directorio /test/monitored.

Someta el siguiente mandato:

```
fteCreateMonitor -ma AGENT_HOP -mm QM_HOP -md /test/monitored
                 -mn myMonitor -mt /home/USER1/task.xml
                 -tr match,*.go -pi 5 -pu minutes
```

3. Un usuario o programa escribe el archivo jump.zip en el directorio /test/monitored y, a continuación, escribe el archivo jump.go en el directorio.
4. El supervisor se desencadena por la existencia del archivo jump.go. El agente sustituye la información sobre el archivo desencadenante en el XML de tarea.

El resultado es la transformación del XML de tarea en:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
```

```

        xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
<managedTransfer>
  <originator>
    <hostName>blue.example.com</hostName>
    <userID>USER1</userID>
  </originator>
  <sourceAgent agent="AGENT_HOP" QMgr="QM_HOP" />
  <destinationAgent agent="AGENT_SKIP" QMgr="QM_SKIP" />
  <transferSet>
    <item mode="binary" checksumMethod="none">
      <source>
        <file>/test/monitored/jump.zip</file>
      </source>
      <destination type="file" exist="overwrite">
        <file>/out/jump.zip</file>
      </destination>
    </item>
  </transferSet>
</managedTransfer>
</request>

```

## Resultados

La transferencia definida por el script XML de tarea se lleva a cabo. AGENT\_HOP lee el archivo `jump.zip` del directorio `/test/monitored` y se transfiere a un archivo denominado `/out/jump.zip` ubicado en el sistema donde se ejecuta AGENT\_SKIP.

### Conceptos relacionados

[“Personalización de las tareas del supervisor de recursos de MFT con sustitución de variables” en la página 249](#)

Cuando se cumplen las condiciones desencadenantes de un supervisor de recursos activo, se llama a la tarea definida. Además de llamar a la tarea de transferencia o mandato con el mismo agente de destino o el mismo nombre de archivo de destino cada vez, también puede modificar la definición de tarea durante la ejecución. Para ello, inserte los nombres de variables en el XML de definición de tarea. Cuando el supervisor determina que se cumplen las condiciones de desencadenante y que la definición de tarea contiene nombres de variable, sustituye los nombres de variable por los valores de variable y, a continuación, llama a la tarea.

### Tareas relacionadas

[“Configuración de tareas de supervisión de MFT para iniciar mandatos y scripts” en la página 243](#)

Los supervisores de recursos no se limitan a realizar transferencias de archivos y la tarea asociada. También puede configurar el supervisor para invocar otros mandatos desde el agente de supervisión, incluidos los programas ejecutables, los scripts Ant o los trabajos JCL. Para llamar a los mandatos, edite el formato XML de definición de tarea del supervisor para incluir uno o varios elementos de mandatos con los parámetros de llamada de mandatos correspondientes, como por ejemplo, argumentos y propiedades.

### Referencia relacionada

**[fteCreateMonitor](#)**: crear un supervisor de recursos de MFT

## Ejemplo: configuración de un recurso MFT

Puede especificar una cola de IBM MQ como el recurso que va a supervisar un supervisor de recursos utilizando el parámetro `-mq` con el mandato **`fteCreateMonitor`**.

### Acerca de esta tarea

En este ejemplo, el recurso que se va a supervisar es la cola `MONITORED_QUEUE`. Esta cola debe estar en el gestor de colas del agente de supervisión, `QM_NEPTUNE`. La condición por la que la cola se supervisa es la presencia de un grupo completo de mensajes. La tarea que se realizará si se cumple la condición se define en el archivo `task.xml`.

**Nota:** No cree más de un supervisor de recursos para supervisar una cola individual. Si lo hace, se puede generar un comportamiento impredecible.



## Procedimiento

Escriba el siguiente mandato:

```
fteCreateMonitor -ma AGENT_NEPTUNE -mn myMonitor -mm QM_NEPTUNE -mq MONITORED_QUEUE  
-mt task.xml -tr completeGroups -pi 5 -pu minutes
```

El supervisor comprueba la cola cada cinco minutos para ver si la condición `completeGroups` es verdadera. Si hay uno o más grupos completos en la cola, el supervisor ejecuta la tarea definida en el archivo `task.xml` una vez para cada grupo completo.

## Personalización de las tareas del supervisor de recursos de MFT con sustitución de variables

Cuando se cumplen las condiciones desencadenantes de un supervisor de recursos activo, se llama a la tarea definida. Además de llamar a la tarea de transferencia o mandato con el mismo agente de destino o el mismo nombre de archivo de destino cada vez, también puede modificar la definición de tarea durante la ejecución. Para ello, inserte los nombres de variables en el XML de definición de tarea. Cuando el supervisor determina que se cumplen las condiciones de desencadenante y que la definición de tarea contiene nombres de variable, sustituye los nombres de variable por los valores de variable y, a continuación, llama a la tarea.



**Atención:** Los nombres de variables no son sensibles a mayúsculas y minúsculas.

Las variables que se utilizan para la sustitución sólo están disponibles para condiciones desencadenantes positivas. Sólo las condiciones desencadenantes `match` y `fileSize` hacen que se sustituyan variables. Si se utiliza una condición `noMatch` y hay nombres de variable de sustitución en la definición de tarea, la tarea no se llama y el supervisor genera un código de retorno de 110 y el mensaje de error BFGDM0060E.

### Si el recurso supervisado es una cola

El valor de cualquier propiedad de mensaje de IBM MQ del primer mensaje que se va a leer de la cola supervisada se puede sustituir en la definición XML de la tarea.

Las propiedades de mensaje definidas por el usuario tienen el prefijo `usr.`, pero no incluya este prefijo en el nombre de variable. Los nombres de variable deben ir precedidos de un carácter de signo de dólar (\$) y escribirse entre llaves, {}.

Por ejemplo, `${destFileName}` se sustituye por el valor de la propiedad de mensaje `usr.destFileName` del primer mensaje que se va a leer de la cola de origen. Si desea más información, consulte [Propiedades de mensaje MQ leídas por MFT de mensajes en colas de origen](#) y [“Supervisión de una cola y utilización de sustitución de variables”](#) en la página 254.

Si una variable no está definida como una propiedad de mensaje, el supervisor notifica un error BFGDM0060E y devuelve el código de retorno 110 (la sustitución de la variable de tarea de supervisor ha fallado). Además, el agente escribe el siguiente mensaje de error en su registro de sucesos (`outputN.log`):

```
BFGDM0113W: Trigger failure for <monitor name> for reason BFGDM0060E: A monitor task could  
not complete as a variable substitution <variable name> was not present.If moderate or verbose  
resource monitor logging is enabled for the monitor, the monitor writes the following message  
to the resource monitor event log of the agent (resmoneventN.log):BFGDM0060E: A monitor task  
could not complete as a variable substitution  
<variable name> was not present.
```

Consulte [“Registro de supervisores de recursos de MFT”](#) en la página 260 para obtener más información sobre el registro del supervisor de recursos.

En la tabla siguiente se muestran las variables de sustitución que se proporcionan de forma predeterminada. Por ejemplo, `${AGENTNAME}` se sustituye por el nombre del agente de supervisor de recursos.

<i>Tabla 15. Variables de sustitución proporcionadas de forma predeterminada</i>	
<b>Variable</b>	<b>Descripción</b>
AGENTNAME	Nombre del agente de supervisor de recursos.
NOMBRECOLA	El nombre de la cola que se está supervisando.
ENCODING	La codificación de caracteres del primer mensaje en la cola o el primer mensaje de un grupo.
MESSAGEID	El ID de mensaje de IBM MQ del primer mensaje de la cola o el primer mensaje del grupo.
GROUPID	El ID de grupo de IBM MQ del grupo o el ID de mensaje si sólo se encuentra un único mensaje. Esta variable sólo se establece si se supervisan grupos completos.
CurrentTimeStamp	Una indicación de fecha y hora basada en la hora local a la que se desencadenó el supervisor. La indicación de fecha y hora es exclusiva para el agente.
CurrentTimeStamp UTC	Una indicación de fecha y hora basada en la hora, en el huso horario UTC, a la que se desencadenó el supervisor. La indicación de fecha y hora es exclusiva para el agente.

### Si el recurso supervisado es un directorio

En la tabla siguiente se muestra el conjunto de nombres de variable que se pueden sustituir en la definición XML de tarea.

<i>Tabla 16. Variables que se pueden sustituir</i>	
<b>Variable</b>	<b>Descripción</b>
FilePath	Nombre completo de la vía de acceso del archivo desencadenante.
FileName	Parte de nombre de archivo del desencadenante.
LastModifiedTime	La hora a la que se modificó por última vez el archivo desencadenante. Esta hora se expresa como la hora local del huso horario en el que se ejecuta el agente y tiene un formato de hora ISO 8601.
LastModifiedDate	La fecha a la que se modificó por última vez el archivo desencadenante. Esta fecha se expresa como fecha local del huso horario en el que se ejecuta el agente y tiene el formato de fecha ISO 8601.
LastModifiedTimeUTC	La hora a la que se modificó por última vez el archivo desencadenante. Esta hora se expresa como la hora local convertida al huso horario UTC y tiene un formato de hora ISO 8601.
LastModifiedDateUTC	La fecha a la que se modificó por última vez el archivo desencadenante. Esta fecha se expresa como fecha local convertida al huso horario UTC y se formatea como fecha ISO 8601.
AgentName	Nombre del agente de supervisor de recursos.
CurrentTimeStamp	Una indicación de fecha y hora basada en la hora local a la que se desencadenó el supervisor. La indicación de fecha y hora es exclusiva para el agente.
CurrentTimeStampUTC	Una indicación de fecha y hora basada en la hora, en el huso horario UTC, a la que se desencadenó el supervisor. La indicación de fecha y hora es exclusiva para el agente.

## Si el recurso supervisado es un archivo desencadenante

La tabla siguiente muestra el conjunto de nombres de variable que se pueden sustituir cuando un supervisor de recursos utiliza el contenido de un archivo desencadenante para determinar los archivos que deben transferirse.

Variable	Descripción
contentSource	El nombre completo de la vía de acceso del archivo de origen.
contentDestination	El nombre completo de la vía de acceso del archivo de destino.

Los nombres de variable deben ir precedidos por un carácter de signo de dólar (\$) e ir entre llaves, {}. Por ejemplo, `${FilePath}` se sustituye por la vía de acceso de archivo completa del archivo desencadenante coincidente.

Existen dos palabras clave especiales que se pueden aplicar a los nombres de las variables para proporcionar un mayor perfeccionamiento. Son las siguientes:

### token

El índice de señales que hay que sustituir (empezando por 1 a la izquierda y empezando por -1 a la derecha)

### separator

Un carácter único para señalar el valor de la variable. El valor predeterminado es el carácter de barra inclinada (/) en plataformas UNIX o el carácter de barra inclinada invertida (\) en plataformas Windows, pero el separador puede ser cualquier carácter válido que pueda aparecer en el valor de la variable.

Si la palabra clave separator se especifica en un nombre de variable, el valor de la variable se divide en señales según el carácter de separador.

El valor que se asigna a la palabra clave token se utiliza como índice para seleccionar qué señal se debe utilizar para sustituir el nombre de variable. El índice de señal es relativo al primer carácter de la variable y se inicia en 1. Si no se especifica la palabra clave de señal, se inserta toda la variable.

Todos los valores que se sustituyen en un nombre de agente en el XML del mensaje se tratan sin distinguir entre mayúsculas y minúsculas. Todos los nombres de Managed File Transfer Agent están en mayúsculas. Si el valor Paris se sustituye en un atributo de agente en el XML del mensaje, este valor se interpreta como una referencia al agente PARIS.

### Conceptos relacionados

[“Ejemplos: Sustitución de variables para definiciones de supervisor de recursos” en la página 251](#)  
Ejemplos de sustitución de variables para definiciones de supervisor de recursos utilizando XML y IBM MQ Explorer.

### Tareas relacionadas

[Qué hacer si la sustitución de variable hace que varios archivos vayan a un solo nombre de archivo](#)

## Ejemplos: Sustitución de variables para definiciones de supervisor de recursos

Ejemplos de sustitución de variables para definiciones de supervisor de recursos utilizando XML y IBM MQ Explorer.

### Ejemplos que muestran cómo funciona la sustitución de variable

Suponiendo que la vía de acceso de archivo al archivo desencadenante coincidente sea `c:\MONITOR\REPORTS\Paris\Report2009.doc` en Windows y `/MONITOR/REPORTS/Paris/`

Report2009.doc en plataformas UNIX , las variables se sustituyen tal como se muestra en la tabla siguiente.

<i>Tabla 18. Cómo se sustituyen las variables</i>	
<b>Especificación de variable</b>	<b>Después de la sustitución de variables</b>
<code>\${FilePath}</code>	Windows :c:\MONITOR\REPORTS\Paris\Report2009.doc UNIX :/MONITOR/REPORTS/Paris/Report2009.doc
<code>\${FilePath{token=1}{separator=.}}</code>	Windows :c:\MONITOR\REPORTS\Paris\Report2009 UNIX :/MONITOR/REPORTS/Paris/Report2009
<code>\${FilePath{token=2}{separator=.}}</code>	Windows : doc UNIX : doc
<code>\${FilePath{token=3}}</code>	Windows : INFORMES UNIX : París

También puede especificar un índice de señales (tokens) negativo para seleccionar señales relativas al último carácter de la variable, como se muestra en la tabla siguiente. Los ejemplos de la tabla utilizan el mismo valor de variable, c:\MONITOR\REPORTS\Paris\Report2009.doc en Windows y /MONITOR/REPORTS/Paris/Report2009.doc en plataformas UNIX .

<i>Tabla 19. Ejemplos de uso de un índice de señales negativo</i>	
<b>Especificación de variable</b>	<b>Después de la sustitución de variables</b>
<code>\${FilePath}</code>	Windows :c:\MONITOR\REPORTS\Paris\Report2009.doc UNIX :/MONITOR/REPORTS/Paris/Report2009.doc
<code>\${FilePath{token=-2}{separator=.}}</code>	Windows :c:\MONITOR\REPORTS\Paris\Report2009 UNIX :/MONITOR/REPORTS/Paris/Report2009
<code>\${FilePath{token=-2}{separator=\}}</code>	Windows : París UNIX : París
<code>\${FilePath{token=-4}}</code>	Windows : SUPERVISOR UNIX : SUPERVISOR

Las variables que se utilizan para la sustitución sólo están disponibles para las siguientes condiciones de desencadenante positivas y la opción noSizeChange, que es una excepción a la regla de condición de desencadenante positiva:

- match
- fileSize
- noSizeChange

Si se utiliza una condición noMatch y hay nombres de variable de sustitución en la definición de tarea, no se llama a la tarea y el supervisor genera un código de retorno de 110 y el mensaje de error BFGDM0060E.

## Ejemplo de uso de XML

El XML de la definición de tarea de ejemplo siguiente utiliza el nombre de agente supervisor como agente de origen para la transferencia (Paris), utiliza el penúltimo nombre de directorio en la vía de acceso de archivo como nombre de agente de destino para la transferencia (Report2009) y renombra el archivo transferido para que sea la raíz del nombre de archivo desencadenante con una extensión de .rpt.

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="{AgentName}" QMgr="QM1" />
    <destinationAgent agent="{FilePath{token=-2}}" QMgr="QMD" />
    <transferSet>
      <item mode="binary" checksumMethod="MD5">
        <source recursive="false" disposition="leave">
          <file>c:/incoming/reports/summary/report.doc</file>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/${FileName{token=1}}{separator=}.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

El resultado es la transformación del XML de tarea en:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT1" QMgr="QM1" />
    <destinationAgent agent="Paris" QMgr="QMD" />
    <transferSet>
      <item mode="binary" checksumMethod="MD5">
        <source recursive="false" disposition="leave">
          <file>c:/incoming/reports/summary/report.doc</file>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/Report2009.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

La variable `{FilePath{token=-2}}` en el atributo `agent` del elemento `<destinationAgent>` se sustituye por el valor `Paris`. Este valor se trata sin distinguir entre mayúsculas y minúsculas y se interpreta como una referencia al agente `PARIS`.

## Ejemplos de uso de IBM MQ Explorer

Al crear un supervisor de recursos mediante IBM MQ Explorer, y una vez que se han especificado las propiedades de supervisor y las condiciones de desencadenante, se proporciona la opción para añadir elementos de transferencia al supervisor. En los ejemplos siguientes se muestra cómo se pueden utilizar las variables `{FilePath}` y `{FileName}` en el **panel Añadir un elemento de transferencia** para personalizar las transferencias que se producen a raíz de una coincidencia de supervisor de recursos.

### Ejemplo 1

Para transferir simplemente el archivo de origen cuando se cumple una condición de desencadenante, se puede utilizar la variable `{FilePath}`:

- Establezca el origen **Nombre de archivo** en `${FilePath}`.
- En el menú desplegable de **Tipo** para el destino, seleccione **Directorio**.
- Establezca el destino **Nombre de archivo** para que sea la ubicación a la que desea que se transfiera el archivo de origen, por ejemplo, podría ser `C:\MFT\out\`.

## Ejemplo 2

Para transferir el archivo de origen a otra ubicación y cambiar la extensión del archivo, se puede utilizar la variable `${FileName}` junto con la variable `${FilePath}`:

En el ejemplo siguiente se presupone que la vía de acceso de archivo del archivo de origen es igual a `C:\MONITOR\REPORTS\Paris\Report2009.doc`:

- Establezca el origen **Nombre de archivo** en `${FilePath}`.
- Establezca el destino **Nombre de archivo** como la ubicación a la que desea que se transfiera el archivo de origen, seguido de `${FileName}{token=1}{separator=.}`, seguido de la nueva extensión del archivo. Por ejemplo, esto podría ser `C:\MFT\out\${FileName}{token=1}{separator=.}.rpt`, lo que equivaldría a `C:\MFT\out\Report2009.rpt` con el nombre de archivo de origen.

## Ejemplo 3

Para utilizar parte de la vía de acceso del archivo de origen para determinar el destino de la transferencia, se puede utilizar la variable `${FilePath}` junto con las especificaciones de señal y separador.

En el ejemplo siguiente se presupone que la vía de acceso del archivo de origen es igual a `C:\MONITOR\REPORTS\Paris\Report2009.doc`.

Es posible utilizar parte de la vía de acceso del archivo de origen para determinar el destino del archivo. Utilizando el ejemplo de vía de acceso de archivo de `C:\MONITOR\REPORTS\Paris\Report2009.doc`, si el archivo se va a transferir a una carpeta en función de la ubicación del archivo de origen, es decir, `Paris` en este ejemplo, se podría realizar lo siguiente:

- Establezca el origen **Nombre de archivo** en `${FilePath}`.
- Establezca el **Nombre de archivo** de destino en el destino donde se sitúan las carpetas para cada ubicación, y luego añada la parte de destino de la vía de acceso y el nombre de archivo. Por ejemplo, esto podría ser `C:\MFT\out\${FilePath}{token=-2}{separator=} \ ${FileName}`, lo que equivaldría a `C:\MFT\out\Paris\Report2009.doc` con el nombre de archivo de origen.

## Conceptos relacionados

[“Personalización de las tareas del supervisor de recursos de MFT con sustitución de variables” en la página 249](#)

Cuando se cumplen las condiciones desencadenantes de un supervisor de recursos activo, se llama a la tarea definida. Además de llamar a la tarea de transferencia o mandato con el mismo agente de destino o el mismo nombre de archivo de destino cada vez, también puede modificar la definición de tarea durante la ejecución. Para ello, inserte los nombres de variables en el XML de definición de tarea. Cuando el supervisor determina que se cumplen las condiciones de desencadenante y que la definición de tarea contiene nombres de variable, sustituye los nombres de variable por los valores de variable y, a continuación, llama a la tarea.

## Tareas relacionadas

[Qué hacer si la sustitución de variable hace que varios archivos vayan a un solo nombre de archivo](#)

## Supervisión de una cola y utilización de sustitución de variables

Puede supervisar una cola y transferir mensajes de la cola supervisada a un archivo utilizando el mandato **ftCreateMonitor**. El valor de cualquier propiedad de mensaje de IBM MQ en el primer mensaje que se va a leer de la cola supervisada se puede sustituir en la definición XML de la tarea y se utiliza para definir el comportamiento de la transferencia.

## Acerca de esta tarea

En este ejemplo, el agente de origen se denomina AGENT\_VENUS, que se conecta a QM\_VENUS. La cola que AGENT\_VENUS supervisa se denomina START\_QUEUE y se encuentra en QM\_VENUS. El agente sondea la cola cada 30 minutos.

Cuando un grupo completo de mensajes se graba en la cola, la tarea de supervisor envía el grupo de mensajes a un archivo en uno de los muchos agentes de destino, que se conectan todos ellos al gestor de colas QM\_MARS. El nombre del archivo al que se transfiere el grupo de mensajes se define mediante la propiedad de mensaje de IBM MQ `usr.fileName` en el primer mensaje del grupo. El nombre del agente al que se envía el grupo de mensajes se define mediante la propiedad de mensaje de IBM MQ `usr.toAgent` en el primer mensaje del grupo. Si la cabecera `usr.toAgent` no está establecida, el valor predeterminado que se va a utilizar para el agente de destino es AGENT\_MAGENTA.

Cuando especifica `useGroups="true"`, si no especifica también `groupId="{GROUPID}"`, la transferencia sólo toma el primer mensaje de la cola. Por ejemplo, si está utilizando la sustitución de variables para generar el nombre de archivo (`fileName`), es posible que el contenido de `a.txt` no sea correcto. Esto se debe a que el nombre de archivo (`fileName`) lo genera el supervisor, pero la transferencia obtiene realmente un mensaje que no es el que debe generar el archivo denominado `fileName`.

## Procedimiento

1. Cree el XML de tarea que define la tarea que el supervisor lleva a cabo cuando se desencadena.

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_VENUS" QMgr="QM_VENUS" />
    <destinationAgent agent="{toAgent}" QMgr="QM_MARS" />
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <queue useGroups="true" groupId="{GROUPID}">START_QUEUE</queue>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/{fileName}.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Las variables que se sustituyen por los valores de las cabeceras de mensajes de IBM MQ se resaltan en **negrita**. Este XML de tarea se guarda en el archivo `/home/USER1/task.xml`

2. Cree un supervisor de recursos para supervisar la cola START\_QUEUE.

Someta el siguiente mandato:

```
fteCreateMonitor -ma AGENT_VENUS -mm QM_VENUS -mq START_QUEUE
                 -mn myMonitor -mt /home/USER1/task.xml
                 -tr completeGroups -pi 30 -pu minutes -dv toAgent=AGENT_MAGENTA
```

3. Un usuario o un programa graba un grupo de mensajes en la cola START\_QUEUE.

El primer mensaje de este grupo cuenta con las siguientes propiedades de mensajes de IBM MQ establecidas:

```
usr.fileName=larmer
usr.toAgent=AGENT_VIOLET
```

4. El supervisor se desencadena cuando se ha grabado el grupo completo. El agente sustituye las propiedades de mensajes de IBM MQ en el XML de tarea.

El resultado es la transformación del XML de tarea en:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_VENUS" QMgr="QM_VENUS" />
    <destinationAgent agent="AGENT_VIOLET" QMgr="QM_MARS" />
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <queue useGroups="true" groupId="{GROUPID}">START_QUEUE</queue>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/larmer.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

## Resultados

Se realiza la transferencia definida por el XML de tarea. El grupo completo de mensajes leídos desde START\_QUEUE por AGENT\_VENUS se escribe en un archivo denominado /reports/larmer.rpt en el sistema donde AGENT\_VIOLET está en ejecución.

## Qué hacer a continuación

### Transferir cada mensaje a un archivo distinto

Si desea supervisar una cola y hacer que cada mensaje se transfiera a un archivo independiente, puede utilizar una técnica similar a la que se ha descrito anteriormente en este tema.

1. Cree el supervisor tal como se ha descrito anteriormente, especificando el parámetro **-tr completeGroups** en el mandato **fteCreateMonitor**.
2. En el XML de la tarea, especifique lo siguiente:

```
<queue useGroups="true" groupId="{GROUPID}">START_QUEUE</queue>
```

Sin embargo, cuando coloque los mensajes en la cola de origen, no los coloque en un grupo de IBM MQ. Añada propiedades de mensaje de IBM MQ a cada mensaje. Por ejemplo, especifique la propiedad `usr.filename` con un valor de nombre de archivo exclusivo para cada mensaje. Esto hace que el Managed File Transfer Agent trate cada mensaje en la cola de origen como un grupo aparte.

## Configuración del comportamiento de reintentos del supervisor para transferencias de mensaje a archivo

Si una transferencia de mensaje a archivo que ha desencadenado un supervisor de recursos falla y deja el grupo de mensajes que desencadenaron el supervisor en la cola, dicha transferencia se vuelve a enviar en intervalos de sondeos posteriores. El número de veces que la transferencia se vuelve a enviar está limitada por la propiedad `monitorGroupRetryLimit` del agente de supervisión.

## Acerca de esta tarea

Cada vez que se desencadena una transferencia de mensaje a archivo, se genera un nuevo ID de transferencia para la tarea de transferencia.



Si se reinicia el agente, el supervisor vuelve a desencadenar una transferencia aunque el número de veces que se ha desencadenado la transferencia haya superado el valor de **monitorGroupRetryLimit** en el archivo `agent.properties`. El valor de la propiedad **monitorGroupRetryLimit** es el número máximo de veces que un supervisor vuelve a desencadenar una transferencia de mensaje a archivo si el grupo de mensajes sigue existiendo en la cola. El valor predeterminado de esta propiedad es 10. El valor de esta propiedad se puede establecer en cualquier valor entero positivo o -1. Si se especifica el valor -1 para esta propiedad, el supervisor vuelve a desencadenar la transferencia un número ilimitado de veces, hasta que no se cumpla la condición desencadenante.

Si un intento de transferencia hace que el número de veces que se ha desencadenado la transferencia supere el valor de **monitorGroupRetryLimit**, el agente escribe un error en el registro de sucesos.

Un solo mensaje se trata como si fuera un solo grupo y la transferencia se vuelve a desencadenar en cada intervalo de sondeo mientras el mensaje permanece en la cola y el número de veces que la transferencia se ha desencadenado sea inferior al valor de **monitorGroupRetryLimit**.

Para establecer la propiedad **monitorGroupRetryLimit** en el agente de supervisión, realice los pasos siguientes:

## Procedimiento

1. Detenga el agente de supervisión utilizando el mandato **fteStopAgent**.
2. Edite el archivo `agent.properties` para que el agente de supervisión incluya la línea siguiente:

```
monitorGroupRetryLimit=number_of_retries
```

El archivo `agent.properties` se encuentra en el directorio `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/monitoring_agent_name`.

3. Inicie el agente de supervisión utilizando el mandato **fteStartAgent**.

## Tareas relacionadas

“Ejemplo: configuración de un recurso MFT” en la página 248

Puede especificar una cola de IBM MQ como el recurso que va a supervisar un supervisor de recursos utilizando el parámetro **-mq** con el mandato **fteCreateMonitor**.

## Utilización de un archivo desencadenante

Puede utilizar el contenido de un archivo desencadenante en un supervisor de recursos para definir un conjunto de archivos que deben transferirse en una única solicitud de transferencia. Cada vez que se detecte un archivo desencadenante coincidente, su contenido se analizará en busca de vías de acceso de archivo de origen y, opcionalmente, vías de acceso de archivo de destino. Estas vías de acceso de archivo se utilizarán para definir elementos del archivo XML de transferencia de tarea que especifique, que se someterá como una única solicitud de transferencia al agente. La definición del supervisor de recursos determina si el contenido del desencadenante está habilitado.

Puede habilitar el desencadenamiento del contenido de archivo cuando cree un supervisor especificando el parámetro **-tc** (contenido de desencadenante). Este parámetro **-tc** se aplica solamente a las opciones de desencadenante de archivo `match` y `noSizeChange`. Si desea más información sobre cómo crear un supervisor, consulte **fteCreateMonitor: crear un supervisor de recursos MFT**.

Cuando se utiliza un archivo de contenido desencadenante, el formato predeterminado de cada línea es:

- Una única vía de acceso de archivo de origen o
- Una vía de acceso de archivo de origen y una vía de acceso de archivo de destino, separadas por una coma

donde los caracteres de espacio en blanco se manejan como parte de las vías de acceso de los archivos. Es posible cambiar el formato de línea predeterminado especificando los parámetros **-tcr** y **-tcc** en el mandato **fteCreateMonitor**. Para obtener más información, consulte “Opciones avanzadas” en la página 259.

Tras analizar un archivo desencadenante, se genera y aplica una lista de vías de acceso de archivo al XML de tarea de transferencia que haya especificado. Tal como ocurre con todos los supervisores, el formato del XML de tarea de transferencia es un XML de tarea de transferencia completo, que genera el mandato **fteCreateTransfer** con un único elemento o archivo definido. El elemento único debe utilizar las variables de sustitución `${contentSource}` y, opcionalmente, `${contentDestination}`, como reemplazo de las vías de acceso de archivo de origen y destino. El supervisor expande el XML de tarea de transferencia para que incluya un elemento de archivo para cada línea (vía de acceso de archivo) del archivo desencadenante.

No puede utilizar el desencadenamiento del contenido con el parámetro **-bs** porque el parámetro **-tc** implica una solicitud de transferencia para cada archivo desencadenante.

## Ejemplo

El ejemplo siguiente define un supervisor para desencadenar en un archivo que finaliza en `trig` y lee las vías de acceso de archivo en dicho archivo.

```
fteCreateTransfer -gt task.xml -sa SrcAgent -da DestAgent -dd /file/destdir ${contentSource}
fteCreateMonitor -mn TrigMonitor -md /home/trigdir -mt task.xml -ma SrcAgent -tr "match,*.trig"
-tc
```

El mandato **fteCreateTransfer** crea un archivo que se denomina `task.xml` para un único archivo, con una vía de acceso de archivo de origen de `${sourceContent}`. Por ejemplo:

```
<item checksumMethod="MD5" mode="binary">
  <source disposition="leave" recursive="false">
    <file>${contentSource}</file>
  </source>
</item>
```

El mandato **fteCreateMonitor** explora en busca de archivos que terminen en `trig`, en el directorio `/home/trigdir`, y utiliza el contenido para crear una única solicitud de transferencia basada en el archivo `task.xml` para todas las vías de acceso de dicho archivo desencadenante. El formato del archivo desencadenante debe ser una vía de acceso de archivo (sólo de origen) en cada línea, sin separador de coma. Por ejemplo:

```
/home/file/first.txt
/home/file/second.txt
/home/different/third.txt
:
```

Todos los archivos se entregan en el directorio `/file/destdir` con su nombre de archivo y no su vía de acceso de archivo, es decir, `/home/file/first.txt` se entrega en `/file/destdir/first.txt`.

De forma alternativa, si cambia el parámetro **-dd /file/destdir** en el mandato **fteCreateTransfer** a **-df \${contentDestination}** y el formato del contenido de un archivo desencadenante a *vía de acceso de archivo de origen, vía de acceso de archivo de destino*, puede definir distintas vías de acceso de destino para el mismo agente de destino. Por ejemplo:

```
/home/file/first.txt,/home/other/sixth.txt
```

La ubicación de destino se convierte en `/home/other/sixth.txt`.

Las variable de sustitución pueden señalizarse. Por ejemplo, puede separar la parte del nombre de archivo de la vía de acceso proporcionada mediante `${contentDestination}{token=-1}`. Por lo tanto, si el destino **fteCreateTransfer** se define como **-df /file/destdir/\${contentDestination}{token=-1}**, el nuevo destino para `/home/file/first.txt` es `/file/destdir/sixth.txt`.

## Opciones avanzadas

Puede cambiar el formato de línea predeterminado para el contenido del archivo desencadenante, mediante el parámetro **-tcr regex**. Proporcione una expresión regular que coincida con el formato de línea pertinente y que proporcione uno o dos grupos de captura. El primer grupo de captura es el origen, y el segundo grupo de captura, opcional, es el destino. Por ejemplo:

- Las vías de acceso de origen y de destino se separan mediante un guión:

```
((?:[^-]+)-((?:[^-]+)+))
```

En este ejemplo, el separador se define en tres ubicaciones, y las tres instancias del guión, -, pueden cambiarse por cualquier otro carácter. Asegúrese de añadir un carácter de escape a los caracteres especiales que aparezcan.

- Las vías de acceso de origen y de destino se separan mediante una coma con espacios al final. Los comentarios que se indiquen mediante un signo de almohadilla (#) se ignoran.

```
((?:[^\, ]+), ((?:[^\, ]+)+) *(?:#.*)+
```

Las vías de acceso de archivo no pueden contener el signo de almohadilla (#). Normalmente, una entrada es la siguiente: `/home/source/from.txt,/home/destination/to.txt # some comment`.

Si utiliza el parámetro **-tcr**, asegúrese de que la expresión regular se haya diseñado y probado bien, de forma que la expresión pueda detectar errores y analizar correctamente los archivos desencadenantes.

Puede invertir el orden de la captura, mediante el parámetro **-tcc destSrc**. Si especifica este parámetro, el primer grupo de captura es la vía de acceso del archivo de destino, y el segundo grupo es la vía de acceso del archivo de origen.

## Cómo se manejan los errores

### Archivo desencadenante vacío

Si el archivo desencadenante está vacío, el resultado es que no se realiza ninguna transferencia de archivos. Es decir, el supervisor crea una solicitud de transferencia pero no se ha especificado ningún elemento de archivo.

### Archivo desencadenante con errores

Si alguna entrada de un archivo desencadenante no puede analizarse en relación al formato esperado, no se generará ninguna solicitud de transferencia. Se publicará un registro de errores del supervisor, y el error también se registrará en el registro de errores. El archivo desencadenante se marca como procesado, y el supervisor no trata de volver a procesar el archivo hasta que se haya actualizado el archivo.

### XML de tarea de transferencia no coincidente

El XML de tarea de transferencia debe coincidir con el archivo desencadenante, es decir, si XML de tarea de transferencia tiene `${sourceContent}` y `${destinationContent}`, todos los archivos desencadenantes de dicho supervisor deben tener vías de acceso de archivo de origen y destino y, de forma parecida, en el caso inverso. En el primer caso, el supervisor informa de que se ha producido un error en la sustitución de `${destinationContent}`, si el archivo desencadenante proporciona sólo la vía de acceso de archivo de origen.

## Ejemplos

El ejemplo siguiente es un desencadenante de contenido básico en que el contenido de un archivo desencadenante tiene sólo una vía de acceso de archivo de origen.

```
fteCreateTransfer -gt task.xml -sa SrcAgent -da DestAgent -dd /file/destdir ${sourceContent}
fteCreateMonitor -mn TrigMonitor -md /home/trigdir -mt task.xml -ma SrcAgent -tr "match,*.trig"
-tc
```

El parámetro **-tcr** define dos grupos de captura de una secuencia de cualquier número de caracteres que estén separados por un carácter de espacio. El parámetro y la opción **-tcc destSrc** indican que los grupos de captura deben procesarse primero como destino y, después, como origen.

```
fteCreateTransfer -gt task.xml -sa SrcAgent -da DestAgent -df ${destinationContent} $
{sourceContent}
fteCreateMonitor -mn TrigMonitor -md /home/trigdir -mt task.xml -ma SrcAgent -tr "match,*.trig"
-tc
    -tcr "((?:[^\ ])+) ((?:[^\ ])+)" -tcc destSrc
```

## V 9.1.0 Registro de supervisores de recursos de MFT

Puede obtener información de diagnóstico sobre supervisores de recursos utilizando el registro.

### Acerca de esta tarea

Puede utilizar el registro para supervisores de recursos utilizando el mandato **fteSetAgentLogLevel** o el archivo `agent.properties` para controlar el registro del supervisor de recursos.

Tenga en cuenta que los puntos de rastreo existentes todavía están en uso para capturar la información.

Los registros del supervisor de recursos se escriben en un archivo denominado `resmoneventN.log`, donde *N* significa un número; por ejemplo, `resmonevent0.log`.



**Atención:** Todos los supervisores de recursos de un agente graban en el mismo archivo de registro.

Para obtener alguna salida de ejemplo de un archivo `resmoneventN.log`, consulte [Qué hacer si el supervisor de recursos de directorio MFT no está desencadenando archivos](#).

En la tabla siguiente se enumeran los tipos de sucesos que el supervisor de recursos graba en el archivo de registro. La tercera columna describe el nivel de registro necesario para capturar cada suceso donde el nivel inferior es INFO y el nivel superior es VERBOSE.

Tenga en cuenta que establecer un nivel de registro elevado, graba también sucesos de nivel inferior. Por ejemplo, al establecer el nivel de registro en MODERATE se graban también los sucesos de nivel INFO, pero no los sucesos de nivel VERBOSE.

Número	Suceso	Nivel de registro	Descripción
1	Supervisor creado	INFO	Se ha creado un supervisor de recursos.
2	Supervisor suprimido	INFO	Se ha suprimido un supervisor de recursos.
3	Supervisor detenido	INFO	Se ha detenido un supervisor de recursos.
4	Supervisor iniciado	INFO	Se ha iniciado un supervisor de recursos.
5	Iniciar sondeo	INFO	Un supervisor de recursos ha iniciado un nuevo ciclo de sondeo.
6	Finalizar sondeo	INFO	Un ciclo de sondeo del supervisor de recursos ha finalizado.
7	Coincidencia de patrón	VERBOSE	Se ha encontrado un archivo en el directorio del supervisor de desencadenante o un mensaje en una cola que coincide con el patrón especificado.
8	Discrepancia de patrón	VERBOSE	Se ha encontrado un archivo en el directorio del supervisor de desencadenante o un mensaje en una cola que no coincide con el patrón especificado.

Número	Suceso	Nivel de registro	Descripción
9	Solicitud de transferencia	INFO	El supervisor de recursos ha iniciado una transferencia.
10	Directorio demasiado profundo	VERBOSE	El directorio supervisado por el supervisor de recursos contiene más subdirectorios que sondear que el número especificado en la configuración del supervisor de recursos.
11	Archivo bloqueado	MODERATE	Otro proceso ha bloqueado el archivo desencadenante supervisado por el supervisor de recursos.
12	Tamaño de archivo pequeño	MODERATE	El archivo desencadenante tiene un tamaño menor que el especificado en la configuración del supervisor de recursos.
13	Tamaño de archivo inestable	MODERATE	El archivo desencadenante cambia con más frecuencia de lo que esperaba la configuración del supervisor de recursos.
14	Demasiados sondeos	MODERATE	Un supervisor de recursos ha sondeado un archivo desencadenante inestable demasiadas veces.
15	Elementos coincidentes	INFO	Número total de archivos desencadenantes encontrados en el directorio sondeado por un supervisor de recursos.
16	Transferir elementos	INFO	Número total de elementos en la solicitud de transferencia.
17	FDC generado	MODERATE	Un supervisor de recursos ha generado una excepción.
18	Solicitud de transferencia	INFO	El supervisor de recursos ha enviado una solicitud de transferencia.
19	Error de inicio de supervisor	MODERATE	No se ha podido iniciar un supervisor de recursos.
20	Historial borrado	INFO	Se ha borrado la información del historial del supervisor.
21	Error al borrar historial del supervisor	INFO	Ha fallado el intento de borrar la información del historial del supervisor.
22	ID de transferencia	INFO	El supervisor ha enviado el ID de la solicitud de transferencia.
23	Proceso por lotes	INFO	Número total de solicitudes de transferencia para elementos coincidentes: $N$ , donde $N$ es un Número.

## Procedimiento

- Para utilizar **fteSetAgentLogLevel** con el fin de activar o desactivar el registro del supervisor de recursos, consulte [fteSetAgentLogLevel](#) para obtener una descripción del parámetro **logMonitor** y ejemplos de cómo utilizar las distintas opciones.

- Para utilizar el archivo `agent.properties` para controlar el registro del supervisor de recursos, consulte El archivo `MFT agent.properties` para obtener una descripción de las propiedades adicionales que le permiten realizar las siguientes actividades de registro:
  - Activar o desactivar el registro
  - Limitar el tamaño de cada archivo de registro
  - Limitar el número de registros que los supervisores de recursos pueden generar

### Ejemplo

El siguiente mensaje de ejemplo establece el registro de nivel `verbose` para el agente HA2, en el gestor de colas MFTDEMO:

```
<?xml version="1.0"?>
<log:log version="6.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xmlns:log="https://www.ibm.com/log">
  <log:originator>
    <log:request>
      <log:hostName>192.168.7.1</log:hostName>
      <log:userID>johndoe</log:userID>
    </log:request>
  </log:originator>
  <log:endpoint agent="HA2" QMgr="MFTDEMO"/>
  <log:logMonitor>MON1="verbose"</log:logMonitor>
</log:log>
```

### Referencia relacionada

[Mandato `fteSetAgentLogLevel`](#)

El archivo `MFT agent.properties`

## Copia de seguridad y restauración de supervisores de recursos de MFT

Puede hacer copia de seguridad de los supervisores de recursos que desea tener disponibles para su uso futuro exportando sus definiciones a un archivo XML que puede importar después para crear un nuevo supervisor de recursos desde la copia de seguridad.

### Acerca de esta tarea

Puede que necesite hacer copia de seguridad de los supervisores de recursos definidos anteriormente para poder volver a utilizar sus definiciones en el futuro, por ejemplo, para volver a crear los supervisores de recursos en otra infraestructura o si se debe crear un supervisor de recursos debido a problemas del gestor de colas.

Puede hacer copia de seguridad de una definición única del gestor de recursos utilizando el mandato **`fteCreateMonitor`** o el mandato **`fteListMonitors`** con el parámetro **`-ox`**. En ambos casos, se hace copia de seguridad de la definición del gestor de recursos exportándola a un archivo XML. Luego se puede utilizar el parámetro **`-ix`** del mandato **`fteCreateMonitor`** para crear un nuevo gestor de recursos importando la definición del archivo XML.

Con el parámetro **`-ox`**, sólo puede realizar una copia de seguridad de una definición de supervisor de recursos a la vez.

**V 9.1.0** En IBM MQ 9.1, el parámetro **`-od`** se añade al mandato **`fteListMonitors`**. Al especificar este parámetro, puede hacer copia de seguridad de más de un supervisor de recursos a la vez exportando sus definiciones de forma masiva a un directorio especificado. Cada definición de supervisor de recursos se guarda en un archivo XML independiente con un nombre en el formato `agent name.monitor name.xml`.

**V 9.1.0** El parámetro **`-od`** es particularmente útil si tiene un gran número de supervisores de recursos de los que desea hacer copia de seguridad porque debe ejecutar el mandato **`fteListMonitors -od`** solo una vez, en lugar de tener que ejecutar el mandato **`fteListMonitors`**

**-ox** por separado para cada definición de recurso o utilizar un script independiente para ejecutar el mandato **ftelistmonitors -ox** para cada supervisor de recursos.

## Procedimiento

- Para hacer copia de seguridad de la definición de un supervisor de recursos exportándolo a un archivo XML, utilice cualquiera de los mandatos siguientes:

- El mandato **ftecreatemonitor** con el parámetro **-ox**.
- El mandato **ftelistmonitors** con el parámetro **-ox**.

Cuando utilice el parámetro **-ox**, también debe especificar los parámetros **-ma** y **-mn**, tal como se muestra en el ejemplo siguiente:

```
ftelistmonitors -ma AGENT1 -mn MONITOR1 -ox filename1.xml
```

### V9.1.0

- Para realizar una copia de seguridad de varias definiciones de supervisor de recursos exportándolas a archivos XML en un directorio especificado, utilice el mandato **ftelistmonitors** con el parámetro **-od** tal como se muestra en el ejemplo siguiente:

```
ftelistmonitors -od /usr/mft/resmonbackup
```

Debe especificar un directorio de destino válido cuando hace copia de seguridad de los supervisores de recursos de forma masiva. Si no se especifica una vía de acceso de destino se genera un mensaje de error como se muestra en el ejemplo siguiente:

```
BFGCL0762E: No se ha especificado el directorio de salida. Vuelva a ejecutar el mandato especificando una vía de acceso válida.
```

El parámetro **-od** no se debe combinar con el parámetro **-ox**; de lo contrario, se visualiza el siguiente mensaje de error:

```
BFGCL0761E: No es válido especificar los dos parámetros '-od' y '-ox' juntos.
```

Puede definir un determinado conjunto de supervisores de recursos para incluir en la copia de seguridad. Por ejemplo, utilizando el parámetro **-ma** para especificar el nombre de un agente, puede realizar una copia de seguridad de todos los supervisores de recursos para dicho agente, tal como se muestra en el ejemplo siguiente:

```
ftelistmonitors -ma AGENT1 -od /usr/mft/resmonbackup
```

Puede utilizar también la coincidencia de comodín incluyendo un carácter de asterisco (\*) al definir un patrón que se va a utilizar para la coincidencia de nombres de agente o nombres de patrón, o ambos. En el ejemplo siguiente se hace copia de seguridad de todos los supervisores de recursos que tienen nombres que coinciden con un patrón especificado y que se encuentran en un agente con un nombre que coincide con un patrón especificado:

```
ftelistmonitors -ma AGENT* -mn MON* -od /usr/mft/resmonbackup
```

Cuando el mandato se ejecuta, se muestran los mensajes de informe de progreso siguientes:

```
Se ha encontrado un total de número definiciones de supervisor de recursos coincidentes.  
Se han guardado índice de número definiciones de supervisor de recursos en el sistema de archivos.
```

Si utiliza la opción de información detallada, el total acumulado se sigue mostrando, pero en lugar de mostrar

```
Se han guardado índice de número definiciones de supervisor de recursos en el sistema de archivos
```

el mandato muestra el nombre de la definición de supervisor guardada, por ejemplo:

```
BFGCL0762I: Definición del supervisor 'FILEMON' del agente 'XFERAGENT' guardada como FILEMON.XFERAGENT.XML al sistema de archivos.
```

### V9.1.0

Para realizar una copia de seguridad de un supervisor de recursos para un agente determinado exportándolo a un archivo XML en un directorio especificado, utilice el mandato **fteListMonitors** con el parámetro **-od** :

```
fteListMonitors -ma AGENT1 -mn MONITOR1 -od /usr/mft/resmonbackup
```

La utilización del parámetro **-od** para realizar una copia de seguridad de un único supervisor de recursos es similar a la utilización del parámetro **-ox**, excepto que el nombre del archivo de salida está en el formato *agent name.monitor name.xml*.

- Para restaurar definiciones de supervisor de recursos de una copia de seguridad, utilice el mandato **fteCreateMonitor** con el parámetro **-ix** como se muestra en el ejemplo siguiente:

```
fteCreateMonitor -ix file name
```

Para obtener más ejemplos de cómo utilizar el parámetro **-od**, consulte [fteListMonitors: listar supervisores de recursos de MFT](#).

### Referencia relacionada

**fteCreateMonitor**: crear un supervisor de recursos de MFT

[fteListMonitors: listar supervisores de recursos de MFT](#)

## V 9.1.3 Borrado del historial del supervisor de recursos

Puede borrar el historial de un supervisor de recursos para que se pueda enviar otra solicitud de transferencia de archivo para un archivo que no se transfirió anteriormente debido a una anomalía. Para borrar el historial del supervisor de recursos, puede utilizar el mandato **fteClearMonitorHistory** o IBM MQ Explorer.

### Antes de empezar

Si la gestión de autorizaciones de usuario se ha habilitado estableciendo el atributo **authorityChecking** en true en el archivo *agent.properties*, el usuario que borra el historial de supervisión debe tener la autorización adecuada, tal como se muestra en la tabla siguiente.

Usuario que borra el historial de supervisión	Autorización de acceso de MFT	Autorización necesaria
El mismo usuario que el que ha creado el supervisor de recursos.	Supervisor	BROWSE on SYSTEM.FTE.AUTHMON1.<monitor_agent_name>  Se trata de la misma autorización que la autorización necesaria para crear o suprimir el supervisor de recursos.
Cualquier usuario que no sea el usuario que ha creado el supervisor de recursos.	Operaciones de supervisión	SET on SYSTEM.FTE.AUTHPS1.<agent_name>  Se trata de la misma autorización que la autorización necesaria para suprimir el supervisor de recursos.

Para obtener más información sobre la gestión de autorizaciones de usuario, consulte [Restricción de las autorizaciones de usuario en las acciones del agente de MFT](#).



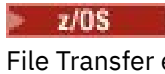
Si un usuario sin la autorización necesaria intenta borrar el historial del supervisor de recursos, el mandato **fteClearMonitorHistory** genera un mensaje de error y registra la anomalía en el archivo `output0.log` del agente. Para obtener más información, consulte [fteClearMonitorHistory: borrar el historial de supervisor de recursos](#).

## Acerca de esta tarea

Si se ha iniciado una transferencia de archivos y no se puede transferir un archivo por alguna razón, el supervisor de recursos no selecciona este archivo para transferirlo de nuevo en su siguiente sondeo porque el historial del supervisor indica que el archivo se ha visto en un sondeo anterior y no se ha modificado desde entonces (consulte [“Conceptos de supervisión de recursos de MFT”](#) en la página 238).

En versiones anteriores a IBM MQ 9.1.3, si un archivo no se puede transferir, la transferencia de archivos solo se puede iniciar de nuevo si el archivo se suprime y luego se vuelve a colocar en el directorio, o si el archivo se actualiza de modo que se modifique el atributo de última fecha de modificación, o si se vuelve a crear el propio supervisor de recursos.

Sin embargo, a partir de IBM MQ 9.1.3, puede borrar el historial del supervisor de recursos utilizando el mandato **fteClearMonitorHistory** o utilizando IBM MQ Explorer. El borrado del historial permite que otra solicitud de transferencia de un archivo que no se ha podido transferir se envíe sin la necesidad de suprimir el archivo y luego volverlo a colocar en el directorio, o actualizar el archivo para cambiar su atributo de última fecha de modificación, que es útil, por ejemplo, en situaciones en las que es necesario transferir el archivo, pero no es posible modificarlo. Poder borrar el historial de un supervisor de recursos significa también que no se tiene que volver a crear el supervisor de recursos para enviar otra solicitud de transferencia de un archivo que no se ha podido transferir.

 A partir de IBM MQ 9.1.3, el miembro SBFGCMD5 de ejemplo que se entrega con Managed File Transfer en z/OS incluye un script de JCL para borrar el historial de un supervisor.

## Procedimiento

- Para utilizar el mandato **fteClearMonitorHistory** para borrar el historial del supervisor de recursos, especifique el mandato en el formato siguiente:

```
fteClearMonitorHistory -p <configuration> -ma <agent name> -mn <monitor name> -w 1000
```

Solo se necesitan los parámetros **-ma** y **-mn**. Los demás parámetros son opcionales. Para obtener más información sobre cómo utilizar el mandato **fteClearMonitorHistory**, incluidos los ejemplos, consulte [fteClearMonitorHistory: clear resource monitor history](#).

Si el historial se borra satisfactoriamente, el mandato genera el siguiente mensaje:

```
BFGCL0780I: Se ha emitido una solicitud para borrar el historial del supervisor de recursos
'nombre de supervisor' del agente 'nombre de agente'.
BFGCL0251I: La solicitud se ha completado satisfactoriamente.
```

y registra el éxito en el archivo `output0.log` del agente.

Si el intento de borrar el historial del supervisor de recursos falla, **fteClearMonitorHistory** genera un mensaje de error y registra la anomalía en el archivo `output0.log` del agente.

- Para utilizar la vista de supervisor de recursos en el plug-in de IBM MQ Explorer MFT para borrar el historial del supervisor de recursos, pulse el botón derecho (del ratón) en el supervisor de recursos y seleccione **Borrar historial** en el menú desplegable.

Si el historial se borra satisfactoriamente, se visualiza el siguiente mensaje:

```
BFGUI00171: El historial del supervisor de recursos se ha borrado correctamente.
```

Si falla el intento de borrar el historial, se visualiza un mensaje de error. Por ejemplo:

```
BFGUI0016E No se ha podido borrar el historial del supervisor de recursos especificado -
motivo 2059.
```

## Cómo trabajar con plantillas de transferencia de archivos

Las plantillas de transferencia de archivos se pueden utilizar para almacenar valores de transferencia de archivos comunes para transferencias repetitivas y complejas. Cree una plantilla de transferencia desde la línea de mandatos utilizando el mandato **fteCreateTemplate** o utilice IBM MQ Explorer para crear una plantilla de transferencia mediante el asistente **Crear nueva plantilla para transferencia de archivos** o guarde una plantilla mientras crea una transferencia de archivos marcando el recuadro de selección **Guardar valores de transferencia como plantilla**. La ventana **Plantillas de transferencia** muestra todas las plantillas de transferencia que ha creado en la red de Managed File Transfer.

### Acerca de esta tarea

Para crear una plantilla de transferencia desde la línea de mandatos, utilice el mandato [fteCreateTemplate](#). A continuación, cuando desee someter una plantilla de transferencia que haya creado en la línea de mandatos, pulse **Someter** en IBM MQ Explorer.

Para ver las plantillas de transferencia en IBM MQ Explorer, siga estos pasos:

### Procedimiento

1. Expanda **Transferencia de archivos gestionada** en la vista Navegador. En la vista Contenido, aparece **Central de transferencias de archivos gestionadas**.
2. En la vista Navegador, aparecen listados todos los gestores de colas de coordinación. Expanda el nombre del gestor de colas de coordinación que ha utilizado para la transferencia planificada. Si desea cambiar el gestor de colas de coordinación al que está conectado, pulse el botón derecho del ratón en el nombre del gestor de colas de coordinación que desea utilizar en la vista Navegador y pulse **Conectar**.
3. Pulse **Plantillas de transferencia**. En la vista Contenido, aparece la ventana **Plantillas de transferencia**.
4. La ventana **Plantillas de transferencia** muestra los siguientes detalles sobre las transferencias de archivos:
  - a) **Nombre** El nombre de la plantilla de transferencia de archivos.
  - b) **Origen** El nombre del agente utilizado para transferir el archivo del sistema de origen.
  - c) **Archivo de origen** El nombre del archivo que se transferirá al sistema host.  
Expanda la información de plantilla de transferencia para ver este campo.
  - d) **Destino** El nombre del agente utilizado para recibir el archivo en el sistema de destino.
  - e) **Archivo de destino** El nombre del archivo de destino después de que se haya transferido al sistema de destino.  
Expanda la información de plantilla de transferencia para ver esta carpeta.
  - f) **Inicio planificado (huso horario seleccionado)** La fecha y hora en que se ha planificado que se inicie la transferencia de archivos, en el huso horario utilizado por el administrador. Para cambiar el huso horario visualizado, pulse **Ventana > Preferencias > IBM MQ Explorer > Managed File Transfer** y seleccione un huso horario alternativo en la lista **Huso horario**. Pulse **Aceptar**.
  - g) **Sucesos desencadenantes** El tipo de suceso que desencadena el inicio de la transferencia de archivos. El tipo puede ser uno de los valores siguientes: existe, no existe o sobrepasa.

### Resultados

Para renovar lo que se visualiza en la ventana **Plantillas de transferencia**, pulse el botón Renovar  en la barra de herramientas de la vista Contenido.

Para someter una plantilla de transferencia e iniciar la transferencia definida en la plantilla, pulse el botón derecho del ratón en el nombre de plantilla y pulse **Someter**.

Para modificar una plantilla de transferencia, pulse el botón derecho del ratón en el nombre de plantilla y pulse **Editar**. Todos los archivos incluidos en la plantilla original aparecen listados como parte de un grupo de transferencias, incluso si no se incluyeron como parte de un grupo en la plantilla original. Si desea eliminar un archivo de la plantilla, debe seleccionar la especificación de archivo en el grupo y pulsar **Eliminar seleccionado**. Si desea añadir nuevas especificaciones de archivo a la plantilla, utilice los campos del panel de plantilla y pulse el botón **Añadir a grupo**. Cuando haya realizado las modificaciones, se le solicitará que asigne un nuevo nombre a la plantilla editada.

Para crear una transferencia de archivos a partir de una plantilla de transferencia, pulse el botón derecho del ratón en el nombre de plantilla y pulse **Editar como nueva transferencia**.

Para crear una copia duplicada de una plantilla de transferencia, pulse el botón derecho del ratón en el nombre de plantilla y pulse **Duplicar**. La plantilla de transferencia duplicada se guarda automáticamente con el mismo nombre que la plantilla original, con la palabra "(copia)" añadida al final.

Para suprimir una plantilla de transferencia, pulse el botón derecho del ratón en el nombre de plantilla y pulse **Suprimir**.

### Tareas relacionadas

[“Creación de una plantilla de transferencia de archivos mediante IBM MQ Explorer” en la página 267](#)

Puede crear una plantilla de transferencia de archivos desde IBM MQ Explorer o desde la línea de mandatos. A continuación, puede utilizar la plantilla para crear nuevas transferencias de archivos utilizando los detalles de plantilla o someter la plantilla para empezar la transferencia de archivos.

### Referencia relacionada

**fteCreateTemplate**: crear nueva plantilla de transferencia de archivos

[fteListTemplates](#)

[fteDeleteTemplates](#)

## Creación de una plantilla de transferencia de archivos mediante IBM MQ Explorer

Puede crear una plantilla de transferencia de archivos desde IBM MQ Explorer o desde la línea de mandatos. A continuación, puede utilizar la plantilla para crear nuevas transferencias de archivos utilizando los detalles de plantilla o someter la plantilla para empezar la transferencia de archivos.

### Acerca de esta tarea

Para crear una transferencia de archivos desde la línea de mandatos, utilice el mandato [fteCreateTemplate](#).

Para crear una plantilla de transferencia de archivos mediante el asistente **Crear nueva plantilla para transferencia de archivos gestionada** en IBM MQ Explorer, ejecute los pasos siguientes:

### Procedimiento

1. En la vista Navegador, pulse **Transferencia de archivos gestionada**. En la vista Contenido, aparece **Central de transferencias de archivos gestionadas**.
2. En la vista Navegador, aparecen todos los gestores de colas de coordinación. Expanda el nombre del gestor de colas de coordinación que ha utilizado para la transferencia planificada. Si desea cambiar el gestor de colas de coordinación al que está conectado, pulse el botón derecho del ratón en el nombre del gestor de colas de coordinación que desea utilizar en la vista Navegador y pulse **Conectar**.
3. Inicie el asistente **Crear nueva plantilla para transferencia de archivos gestionada** pulsando con el botón derecho del ratón en **Plantillas de transferencia** y a continuación, pulse **Nueva plantilla**.
4. Siga las instrucciones de los paneles del asistente. Para cada panel se proporciona ayuda según contexto. Para acceder a la ayuda según contexto en Windows, pulse F1. En Linux, pulse **Ctrl+F1** o **Shift+F1**.

Si ha creado una plantilla que contiene todos los detalles de la transferencia necesarios, asegúrese de que ha seleccionado el recuadro de selección **Guardar valores de transferencia como plantilla**

en la página **Resumen de la transferencia** si este recuadro de selección aún no está seleccionado. Asimismo, especifique un nombre para la plantilla en el campo Nombre. Si crea una plantilla que aún no contiene todos los detalles de la transferencia necesarios, el recuadro de selección **Guardar valores de transferencia como plantilla** se marcará automáticamente.

### Tareas relacionadas

“Cómo trabajar con plantillas de transferencia de archivos” en la página 266

Las plantillas de transferencia de archivos se pueden utilizar para almacenar valores de transferencia de archivos comunes para transferencias repetitivas y complejas. Cree una plantilla de transferencia desde la línea de mandatos utilizando el mandato **fteCreateTemplate** o utilice IBM MQ Explorer para crear una plantilla de transferencia mediante el asistente **Crear nueva plantilla para transferencia de archivos** o guarde una plantilla mientras crea una transferencia de archivos marcando el recuadro de selección **Guardar valores de transferencia como plantilla**. La ventana **Plantillas de transferencia** muestra todas las plantillas de transferencia que ha creado en la red de Managed File Transfer.

### Referencia relacionada

**fteCreateTemplate**: crear nueva plantilla de transferencia de archivos

[fteListTemplates](#)

[fteDeleteTemplates](#)

## Copia de seguridad de una definición de plantilla de transferencia de archivos

Las plantillas de transferencia de archivos contienen un documento XML que define las especificaciones de archivo de origen y destino para la transferencia. Puede utilizar este archivo XML como entrada para el mandato **fteCreateTemplate** para volver a crear una plantilla de transferencia de archivos.

### Acerca de esta tarea

Para realizar copia de seguridad del documento XML que contiene las especificaciones de archivo de origen y destino de una plantilla de transferencia, utilice el mandato [fteCreateTransfer](#) o IBM MQ Explorer. Para crear un archivo de copia de seguridad con formato XML de plantilla de transferencia, siga estos pasos:

### Procedimiento

- Método uno: utilice el parámetro **-gt** en un mandato [fteCreateTransfer](#) para generar un mensaje XML de plantilla de transferencia a un archivo nuevo.
- Método dos: cree la plantilla utilizando IBM MQ Explorer.  
Cuando llegue a la página *Resumen de la plantilla de la transferencia*:
  - a) Copie *Vista previa XML de mensaje de solicitud*.
  - b) Guarde este mensaje XML de plantilla de transferencia en un archivo nuevo.
- Método tres: utilice IBM MQ Explorer para hacer copia de seguridad de las plantillas existentes.
  - a) Vaya a **Managed File Transfer > Nombre del gestor de colas > Plantillas de transferencia**.
  - b) En el panel de transferencia, resalte la plantilla de la que se debe hacer una copia de seguridad, pulse el botón derecho (del ratón) y seleccione **Editar** en el menú emergente.
  - c) Pulse **Siguiente** hasta que llegue a la página *Resumen de la plantilla de la transferencia*.
  - d) Copie *Vista previa XML de mensaje de solicitud*.
  - e) Guarde este mensaje XML de plantilla de transferencia en un archivo nuevo.

### Resultados

Puede utilizar el archivo de mensaje XML de plantilla de transferencia, creado mediante uno de los métodos anteriores, como entrada para el mandato [fteCreateTemplate](#). Consulte el mandato **fteCreateTemplate** para obtener detalles de cómo utilizar este mandato.

## Referencia relacionada

[Mandato fteCreateTemplate](#)

[Mandato fteListTemplates](#)

## Transferencia de datos de archivos a mensajes

Puede utilizar la característica de archivo a mensaje de Managed File Transfer para transferir datos desde un archivo a un único mensaje o varios mensajes, en una cola de IBM MQ.

Para realizar transferencias de archivo a mensaje y de mensaje a archivo, tanto el agente de origen como el de destino de la transferencia deben ser IBM WebSphere MQ 7.5, o posterior, o IBM WebSphere MQ File Transfer Edition 7.0.3, o posterior. Para obtener información sobre transferencias de mensaje a archivo, consulte [“Transferir datos de mensajes a archivos”](#) en la [página 277](#).

El agente de destino para la transferencia de archivo a mensaje no puede ser un agente de puente de protocolo o un agente de puente Connect:Direct.

Puede transferir datos de archivo a datos de mensaje de IBM MQ. Los mensajes de IBM MQ pueden ser leídos y utilizados por las aplicaciones. Los siguientes tipos de transferencias de archivo a mensaje está soportados:

- De un único archivo a un único mensaje. El mensaje no tiene establecido un ID de grupo de IBM MQ.
- De un único archivo a varios mensajes, dividiendo el archivo en mensajes de una longitud determinada. Los mensajes tienen todos el mismo ID de grupo de IBM MQ.
- De un único archivo a varios mensajes, dividiendo un archivo de texto en un delimitador de expresión regular Java. Los mensajes tienen todos el mismo ID de grupo de IBM MQ.
- De un único archivo hasta varios mensajes, dividiendo un archivo binario en un delimitador hexadecimal. Los mensajes tienen todos el mismo ID de grupo de IBM MQ.

Si desea dividir un archivo binario utilizando una secuencia de bytes como delimitador, utilice el parámetro **-sqdb** del mandato **fteCreateTransfer**. Para obtener más información, consulte el [parámetro -sqdb](#).

De forma predeterminada los mensajes creados por una transferencia de archivo a mensaje son persistentes. Los mensajes pueden configurarse para ser no persistente o para tener el valor de persistencia definido por la cola de destino.

Si especifica que un archivo se divida en varios mensajes, todos los mensajes creados desde el archivo tendrán el mismo ID de grupo de IBM MQ. Si no especifica que un archivo se divida en varios mensajes, sólo se creará un mensaje desde el archivo y este mensaje no tendrá establecido el ID de grupo de IBM MQ.

Si va a transferir archivos a mensajes grandes, o a muchos mensajes pequeños, es posible que tenga que cambiar algunas propiedades de IBM MQ o Managed File Transfer. Si desea más información al respecto, consulte [Instrucciones para establecer atributos MQ y propiedades MFT asociadas al tamaño del mensaje](#).

**Nota:** Si la cola de destino es una cola en clúster o un alias para una cola en clúster, obtendrá un mensaje de error al transferir un archivo en una cola si la propiedad de agente `enableClusterQueueInputOutput` no se ha establecido en `true`. Si desea más información, consulte [Qué hacer si la cola de destino es una cola agrupada en clúster, o un alias a una cola agrupada en clúster](#)

## Configuración de un agente para realizar transferencias de archivo a mensaje

De forma predeterminada, los agentes pueden realizar transferencias de archivo a mensaje o de mensaje a archivo. Para habilitar esta función debe establecer la propiedad de agente `enableQueueInputOutput` en `true` (verdadera). Para habilitar la escritura en las colas en clúster de IBM MQ, también debe establecer la propiedad de agente `enableClusterQueueInputOutput` en `true`.

## Acerca de esta tarea

Si intenta realizar una transferencia de archivo a mensaje a un agente de destino que no tenga la propiedad `enableQueueInputOutput` establecida en `true`, la transferencia fallará. El mensaje de registro de transferencia publicado en el gestor de cola de coordinación contiene el siguiente mensaje:

```
BFGI00197E: An attempt to write to a queue was rejected by the destination agent. The agent must have enableQueueInputOutput=true set in the agent.properties file to support transferring to a queue.
```

Para que el agente pueda escribir y leer en las colas, siga los siguientes pasos:

## Procedimiento

1. Detenga el agente de destino utilizando el mandato **fteStopAgent**.
2. Edite el archivo `agent.properties` para incluir la línea `enableQueueInputOutput=true`.  
El archivo `agent.properties` se encuentra en el directorio `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/destination_agent_name`.
3. Opcional: Edite el archivo `agent.properties` para incluir la línea `enableClusterQueueInputOutput=true`. El archivo `agent.properties` se encuentra en el directorio `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/destination_agent_name`.
4. Inicie el agente de destino utilizando el mandato **fteStartAgent**.

## Ejemplo: Transferencia de un único archivo en un mensaje

Puede especificar una cola como destino de una transferencia de archivo utilizando el parámetro **-dq** con el mandato **fteCreateTransfer**. El archivo de origen debe ser más pequeño que la longitud máxima de mensaje establecida en la cola de destino. La cola de destino no tiene que estar en el mismo gestor de colas que el gestor de colas al que se conecta el agente de destino, pero estos dos gestores de colas deben poder comunicarse.

## Acerca de esta tarea

El archivo de origen se denomina `/tmp/single_record.txt` y se encuentra en el mismo sistema que el agente de origen, `AGENT_NEPTUNE`. El agente de origen, `AGENT_NEPTUNE`, utiliza el gestor de colas `QM_NEPTUNE`. El agente de destino es `AGENT_VENUS` y este agente se conecta con el gestor de colas `QM_VENUS`. La cola de destino `RECEIVING_QUEUE`, se encuentra en el gestor de colas `QM_MERCURY`. `QM_MERCURY` se encuentra en la misma red de IBM MQ que el gestor de colas `QM_VENUS` al que puede acceder.

## Procedimiento

Escriba el siguiente mandato:

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_VENUS -dm QM_VENUS  
-dq RECEIVING_QUEUE@QM_MERCURY /tmp/single_record.txt
```

Si la cola de destino se encuentra en un gestor de colas diferente al gestor de colas utilizado por el agente de destino, debe especificar el valor del parámetro **-dq** en el formato siguiente `nombre_cola@nombre_gestor_colas`. Si no especifica `@nombre_gestor_colas` en el valor, el agente de destino da por supuesto que la cola de destino se encuentra en el gestor de colas del agente de destino. Se produce la excepción cuando la propiedad de agente `enableClusterQueueInputOutput` se ha establecido en `true`. En este caso, el agente de destino utilizará procedimientos de resolución de IBM MQ estándar para determinar dónde se encuentra la cola.

El agente de origen, `AGENT_NEPTUNE`, lee los datos del archivo `/tmp/single_record.txt` y transfiere estos datos al agente de destino, `AGENT_VENUS`. El agente de destino, `AGENT_VENUS`, envía los datos a un mensaje persistente en la cola `RECEIVING_QUEUE@QM_MERCURY`. El mensaje no tiene establecido un ID de grupo de IBM MQ.

## Ejemplo: División de un único archivo en varios mensajes por longitud

Puede dividir un archivo en varios mensajes de IBM MQ utilizando el parámetro **-qs** del mandato **fteCreateTransfer**. El archivo se divide en secciones de longitud fija, cada una de las cuales se escribe en un mensaje individual.

### Acerca de esta tarea

El archivo de origen se denomina `/tmp/source.file` y tiene 36 KB de tamaño. El archivo de origen se encuentra en el mismo sistema que el archivo de origen `AGENT_NEPTUNE`. El agente de origen, `AGENT_NEPTUNE`, se conecta con el gestor de colas `QM_NEPTUNE`. El agente de destino es `AGENT_MERCURY`, que se conecta con el gestor de colas `QM_MERCURY`. La cola de destino `RECEIVING_QUEUE`, también se encuentra en el gestor de colas `QM_MERCURY`. La transferencia divide el archivo de origen en secciones de 1 KB y escribe cada una de estas secciones en un mensaje en `RECEIVING_QUEUE`.

### Procedimiento

Escriba el siguiente mandato:

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_MERCURY -dm QM_MERCURY  
-dq RECEIVING_QUEUE -qs 1K /tmp/source.file
```

El agente de origen, `AGENT_NEPTUNE`, lee los datos del archivo `/tmp/source.file` y los transfiere al agente de destino, `AGENT_MERCURY`. El agente de destino, `AGENT_MERCURY`, escribe los datos en tres y seis mensajes persistentes de 1KB en la cola `RECEIVING_QUEUE@QM_MERCURY`. Estos mensajes tienen todos el mismo ID de grupo de IBM MQ; el último mensaje del grupo tiene el distintivo de IBM MQ `LAST_MSG_IN_GROUP` definido.

## Ejemplo: división un archivo de texto en varios mensajes utilizando un delimitador de expresión regular

Transferencia de un único archivo de texto con varios mensajes dividiendo el archivo en cada coincidencia de una expresión regular Java determinada. Para hacerlo, utilizará el parámetro **-dqdt** del mandato **fteCreateTransfer**.

### Acerca de esta tarea

El archivo se divide en secciones de longitud variable, cada una de las cuales se escribe en un mensaje individual. El archivo de texto se divide en cada punto en que el texto en el archivo coincide con una expresión regular determinada. El archivo de origen se llama `/tmp/names.text` y tiene el contenido siguiente:

```
Jenny Jones,John Smith,Jane Brown
```

La expresión regular que especifica dónde dividir el archivo es un carácter de coma (,).

El archivo de origen se encuentra en el mismo sistema del agente de origen `AGENT_NEPTUNE`, que se conecta con el gestor de colas `QM_NEPTUNE`. La cola de destino `RECEIVING_QUEUE`, se encuentra en el gestor de colas `QM_MERCURY`. `QM_MERCURY` es también el gestor de colas utilizado por el agente de destino `AGENT_MERCURY`. La transferencia divide el archivo de origen en secciones y escribe cada una de estas secciones en un mensaje en `RECEIVING_QUEUE`.

### Procedimiento

Escriba el siguiente mandato:

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_MERCURY -dm QM_MERCURY
-dq RECEIVING_QUEUE -t text -dqdp postfix -dqdt "," /tmp/names.text
```

El agente de origen, AGENT\_NEPTUNE, lee los datos del archivo /tmp/names.text y transfiere estos datos al agente de destino, AGENT\_MERCURY. El agente de destino, AGENT\_MERCURY, escribe los datos en tres mensajes persistentes en la cola *RECEIVING\_QUEUE*. Estos mensajes tienen todos el mismo ID de grupo de IBM MQ; el último mensaje del grupo tiene el distintivo de IBM MQ *LAST\_MSG\_IN\_GROUP* definido.

Los datos en los mensajes son los siguientes.

- Primer mensaje:

```
Jenny Jones
```

- Segundo mensaje:

```
John Smith
```

- Tercer mensaje:

```
Jane Brown
```

## Ejemplo: División de un archivo de texto con un delimitador de expresión regular e incluyendo el delimitador en los mensajes

Transferencia de un único archivo de texto en varios mensajes dividiendo el archivo en cada coincidencia de una expresión regular Java determinada e inclusión de la coincidencia de la expresión regular en los mensajes resultantes. Para hacerlo, utilice los parámetros **-dqdt** y **-qi** del mandato **fteCreateTransfer**.

### Acerca de esta tarea

Transfiera un único archivo de texto en varios mensajes en una cola. El archivo se divide en secciones de longitud variable, cada una de las cuales se escribe en un mensaje individual. El archivo de texto se divide en cada punto en que el texto en el archivo coincide con una expresión regular determinada. El archivo de origen se llama /tmp/customers.text y tiene el contenido siguiente:

```
Customer name: John Smith
Customer contact details: john@example.net
Customer number: 314

Customer name: Jane Brown
Customer contact details: jane@example.com
Customer number: 42

Customer name: James Jones
Customer contact details: jjones@example.net
Customer number: 26
```

La expresión regular que especifica dónde dividir el archivo es `Customer\number:\s\d+`, que coincide con el texto "Customer number: " seguido de cualquier número de dígitos. Las expresiones regulares especificadas en la línea de mandatos deben rodearse de comillas dobles para impedir que el shell de mandatos evalúe la expresión regular. La expresión regular se evalúa como expresión regular Java. Si desea más información, consulte [Expresiones regulares utilizadas por MFT](#).

De forma predeterminada, el número de caracteres con el que puede coincidir una expresión regular está establecido en cinco. La expresión regular utilizada en este ejemplo coincide con las series que sean mayores de cinco caracteres. Para permitir coincidencias mayores de cinco caracteres, edite el archivo de propiedades del agente para incluir la propiedad **maxDelimiterMatchLength**.



De forma predeterminada, el texto que coincide con la expresión regular no se incluye en los mensajes. Para incluir el texto que coincide con la expresión regular en los mensajes, como en este ejemplo, utilice el parámetro **-qi**. El archivo de origen se encuentra en el mismo sistema del agente de origen AGENT\_NEPTUNE, que se conecta con el gestor de colas QM\_NEPTUNE. La cola de destino RECEIVING\_QUEUE, se encuentra en el gestor de colas QM\_MERCURY. QM\_MERCURY es también el gestor de colas utilizado por el agente de destino AGENT\_MERCURY. La transferencia divide el archivo de origen en secciones y escribe cada una de estas secciones en un mensaje en RECEIVING\_QUEUE.

## Procedimiento

1. Detenga el agente de destino utilizando el siguiente mandato:

```
fteStopAgent AGENT_MERCURY
```

2. Añada la siguiente línea al archivo de propiedades del agente para AGENT\_MERCURY:

```
maxDelimiterMatchLength=25
```

**Nota:** Aumentar el valor de **maxDelimiterMatchLength** puede reducir el rendimiento.

3. Inicie el agente de destino utilizando el siguiente mandato:

```
fteStartAgent AGENT_MERCURY
```

4. Escriba el siguiente mandato:

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_MERCURY -dm QM_MERCURY  
-dq RECEIVING_QUEUE  
text -dqdt "Customer\snumber:\s\d+" -qi -dqdp postfix /tmp/customers.text
```

El agente de origen, AGENT\_NEPTUNE, lee los datos del archivo /tmp/customers.text y transfiere estos datos al agente de destino, AGENT\_MERCURY. El agente de destino, AGENT\_MERCURY, escribe los datos en tres mensajes persistentes en la cola RECEIVING\_QUEUE. Estos mensajes tienen todos el mismo ID de grupo de IBM MQ; el último mensaje del grupo tiene el distintivo de IBM MQ LAST\_MSG\_IN\_GROUP definido.

Los datos en los mensajes son los siguientes.

- Primer mensaje:

```
Customer name: John Smith  
Customer contact details: john@example.net  
Customer number: 314
```

- Segundo mensaje:

```
Customer name: Jane Brown  
Customer contact details: jane@example.com  
Customer number: 42
```

- Tercer mensaje:

```
Customer name: James Jones  
Customer contact details: jjones@example.net  
Customer number: 26
```

## Ejemplo: establecer propiedades de mensaje de IBM MQ en una transferencia de archivo a mensaje

Puede utilizar el parámetro **-qmp** en el mandato **fteCreateTransfer** para especificar si las propiedades de mensaje de IBM MQ han sido establecidas por la transferencia en el primer mensaje escrito en la cola de destino. Las propiedades de mensaje de IBM MQ permiten que una aplicación seleccione mensajes que procesar o que recupere información sobre un mensaje sin acceder a cabeceras de IBM MQ Message Descriptor (MQMD) o MQRFH2.

### Acerca de esta tarea

Incluya el parámetro `-qmp true` en el mandato **fteCreateTransfer**. En este ejemplo, el ID de usuario MQMD del usuario que envía el mandato es `larmer`.

### Procedimiento

Escriba el siguiente mandato:

```
fteCreateTransfer -sa AGENT_JUPITER -da AGENT_SATURN -dq MY_QUEUE@MyQM -qmp true
-t text /tmp/source_file.txt
```

Las propiedades de mensaje de IBM MQ del primer mensaje escruo por el agente de destino, `AGENT_SATURN`, en la cola, `MY_QUEUE`, en el gestor de colas, `MyQM`, están definidas con estos valores:

```
usr.WMQFTETransferId=414cbaedefa234889d999a8ed09782395ea213ebbc9377cd
usr.WMQFTETransferMode=text
usr.WMQFTESourceAgent=AGENT_JUPITER
usr.WMQFTEDestinationAgent=AGENT_SATURN
usr.WMQFTEFileName=source_file.txt
usr.WMQFTEFileSize=1024
usr.WMQFTEFileLastModified=1273740879040
usr.WMQFTEFileIndex=0
usr.WMQFTEMqmdUser=larmer
```

## Ejemplo: Configuración de las propiedades definidas por el usuario en una transferencia de archivo a mensaje

Los metadatos definidos por el usuario se establecen como una propiedad de mensaje de IBM MQ en el primer mensaje que la transferencia ha grabado en la cola de destino. Las propiedades de mensaje de IBM MQ permiten que una aplicación seleccione mensajes que procesar o que recupere información sobre un mensaje sin acceder a las cabeceras de IBM MQ Message Descriptor (MQMD) o MQRFH2.

### Acerca de esta tarea

Incluya los parámetros `-qmp true` y `-md account=123456` en el mandato **fteCreateTransfer**, para establecer la propiedad `usr.account` en `123456` en la cabecera RFH2.

### Procedimiento

Escriba el siguiente mandato:

```
fteCreateTransfer -sa AGENT_JUPITER -da AGENT_SATURN -dq MY_QUEUE@MyQM
-qmp true -md account=123456 /tmp/source_file.txt
```

Además del conjunto estándar de propiedades de mensajes de IBM MQ, la propiedad definida por el usuario está establecida en la cabecera de mensaje del primer mensaje escrito por el agente de destino,

AGENT\_SATURN, en la cola, MY\_QUEUE, en el gestor de colas, MyQM. La cabecera se establece en el siguiente valor:

```
usr.account=123456
```

El prefijo `usr` se añade al principio del nombre de los metadatos definidos por el usuario.

### ***Ejemplo: añadir una propiedad de mensaje definida por el usuario para una transferencia de archivo a mensaje***

Si utiliza Managed File Transfer para las transferencias gestionadas de mensaje a archivo, puede incluir una propiedad de mensaje definida por el usuario para el mensaje resultante.

## **Acerca de esta tarea**

Puede utilizar cualquiera de los métodos siguientes para definir una propiedad de mensaje personalizada:

- Especifique el parámetro **-md** en la solicitud de transferencia. Para obtener más información, consulte [“Ejemplo: Configuración de las propiedades definidas por el usuario en una transferencia de archivo a mensaje”](#) en la página 274.
- Utilice una tarea Ant; puede utilizar `fte:filecopy` o `fte:filemove`. En el ejemplo siguiente se muestra una tarea `fte:filecopy`:

```
<project xmlns:fte="antlib:com.ibm.wmqfte.ant.taskdefs" default="complete">
<!-- Initialise the properties used in this script.-->

<target name="init" description="initialise task properties">
    <property name="src.file" value="/home/user/file1.bin"/>
    <property name="dst.queue" value="TEST.QUEUE@qm2"/>
    <fte:uuid property="job.name" length="8"
prefix="copyjob#"/>
</target>
<target name="step1" depends="init" description="transfer file">

<fte:filecopy cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
src="agent1@qm1" dst="agent2@qm2"
rcproperty="copy.result">

<fte:metadata>
<fte:entry name="fileName" value="{FileName}"/>
</fte:metadata>

<fte:filespec srcfilespec="{src.file}" dstqueue="{dst.queue}"
dstmsgprops="true"/>

</fte:filecopy>

</target>
</project>
```

- Utilizar un supervisor de recursos y una sustitución de variable. En el ejemplo siguiente se muestra parte de un XML de tarea de transferencia:

```
<?xml version="1.0" encoding="UTF-8"?>
<monitor:monitor
xmlns:monitor="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance" version="5.00"
xsi:schemaLocation="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinit ion ./Monitor.xsd">
<name>METADATA</name>
<pollInterval units="minutes">5</pollInterval>
<batch maxSize="5"/>
<agent>AGENT1</agent>
<resources>
<directory recursionLevel="0">e:\temp</directory>
</resources>
<triggerMatch>
<conditions>
<allof>
```

```

    <condition>
      <fileMatch>
        <pattern>*.txt</pattern>
      </fileMatch>
    </condition>
  </allOf>
</conditions>
</triggerMatch>
<tasks>
  <task>
    <name/>
    <transfer>
      <request version="5.00"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
        <managedTransfer>
          <originator>
            <hostName>mqjason.raleigh.ibm.com.</hostName>
            <userID>administrator</userID>
          </originator>
          <sourceAgent QMgr="AGENTQM" agent="AGENT1"/>
          <destinationAgent QMgr="AGENTQM" agent="AGENT2"/>
          <transferSet priority="0">
            <metaDataSet>
              <metaData key="FileName">${FileName}</metaData>
            </metaDataSet>
            <item checksumMethod="MD5" mode="text">
              <source disposition="delete" recursive="false">
                <file>${FilePath}</file>
              </source>
              <destination type="queue">
                <queue persistent="true"
setMqProps="true">TEST.QUEUE@AGENTQM</queue>
              </destination>
            </item>
          </transferSet>
        </managedTransfer>
      </request>
    </transfer>
  </task>
</tasks>
<originator>
  <hostName>mqjason.raleigh.ibm.com.</hostName>
  <userID>administrator</userID>
</originator>
</monitor:monitor>

```

### Tareas relacionadas

[“Ejemplo: establecer propiedades de mensaje de IBM MQ en una transferencia de archivo a mensaje” en la página 274](#)

Puede utilizar el parámetro **-qmp** en el mandato **fteCreateTransfer** para especificar si las propiedades de mensaje de IBM MQ han sido establecidas por la transferencia en el primer mensaje escrito en la cola de destino. Las propiedades de mensaje de IBM MQ permiten que una aplicación seleccione mensajes que procesar o que recupere información sobre un mensaje sin acceder a cabeceras de IBM MQ Message Descriptor (MQMD) o MQRFH2.

### Referencia relacionada

[Tarea Ant fte:filecopy](#)

[Tarea Ant fte:filemove](#)

## Fallo en una transferencia de archivo a mensaje

Si una transferencia de archivo a mensaje falla después de que el agente haya empezado a escribir los datos de archivo en la cola de destino, el agente escribe un mensaje en la cola para indicar a la aplicación que consume los mensajes que se ha producido un fallo.

El mensaje escrito en la cola de destino si se produce un fallo:

- está vacío

- tiene el mismo ID de grupo de IBM MQ que el mensaje anterior escrito en la cola de destino por el agente
- tiene establecido el distintivo de IBM MQ LAST\_MSG\_IN\_GROUP
- contiene propiedades de mensaje adicionales de IBM MQ, si se habilitan las propiedades del mensaje. Si desea más información, consulte el tema [Propiedades de mensaje MQ establecidas por MFT en mensajes escritos en colas de destino](#).

**V 9.1.0.8** A partir de IBM MQ 9.1.0 Fix Pack 8, cuando falla una transferencia desde un archivo, debido a un error de comprobación de tamaño de delimitador, sólo se envía un mensaje vacío. Además, las propiedades de usuario se añaden a este mensaje si la causa del error de transferencia era que el delimitador sobrepasaba el tamaño establecido en el agente de destino.

## Ejemplo

Se solicita una transferencia ejecutando el siguiente mandato:

```
fteCreateTransfer -sa AGENT_JUPITER -da AGENT_SATURN -dq RECEIVING_QUEUE
-qmp true -qs 1K /tmp/source1.txt
```

El archivo `source1.txt` es de 48 KB. La transferencia divide este archivo en mensajes de 1 KB y escribe dichos mensajes en la cola de destino `RECEIVING_QUEUE`.

Mientras la transferencia está en curso, después de que el agente ha escrito 16 mensajes en `RECEIVING_QUEUE`, se produce un fallo en el agente de origen.

El agente escribe un mensaje en blanco en `RECEIVING_QUEUE`. Además del conjunto estándar de propiedades de mensaje, el mensaje en blanco tiene las siguientes propiedades definidas:

```
usr.WMQFTEResultCode = 40
usr.WMQFTESupplement = BFGTR0036I: The transfer failed to complete successfully.
```

## Transferir datos de mensajes a archivos

La función de mensaje a archivo de Managed File Transfer le permite transferir datos de uno o más mensajes en una cola de IBM MQ a un archivo, un conjunto de datos (en z/OS) o un espacio de archivos de usuario. Si tiene una aplicación que crea o procesa mensajes de IBM MQ, puede utilizar la función de mensaje a archivo de Managed File Transfer para transferir estos mensajes a un archivo en cualquier sistema de la red de Managed File Transfer.

Para obtener información sobre transferencias de mensaje a archivo, consulte [“Transferencia de datos de archivos a mensajes”](#) en la página 269.



**Atención:** El agente de origen para la transferencia de mensaje a archivo no puede ser un agente de puente de protocolo o un agente de puente Connect:Direct.

Puede transferir datos de mensajes de IBM MQ a un archivo. Se da soporte a los tipos siguientes de transferencias de mensaje a archivo:

- De un único mensaje a un único archivo
- De varios mensajes a un único archivo
- De varios mensajes con el mismo ID de grupo de IBM MQ a un único archivo.
- De varios mensajes a un único archivo, incluyendo un delimitador de texto o binario entre los datos de cada mensaje escrito en el archivo.

Si va a transferir archivos desde mensajes grandes, o desde muchos mensajes pequeños, es posible que tenga que cambiar algunas propiedades de IBM MQ o Managed File Transfer. Para obtener más información sobre, consulte [Guía para establecer los atributos de MQ y las propiedades de MFT asociadas con el tamaño del mensaje](#).

## V 9.1.0

Desde IBM MQ 9.1.0, en un mensaje a la transferencia de archivos, el agente de origen examina los mensajes de la cola de origen, a diferente del GET destructivo en versiones anteriores de IBM MQ. Los mensajes se eliminan de la cola de origen después de examinar todos los mensajes (de un grupo si se utiliza la agrupación de mensajes) y tras grabar los datos en el archivo de destino. Esto permite que los mensajes permanezcan en la cola de origen si falla o se cancela una transferencia. Debido a este cambio, también se debe proporcionar autoridad BROWSE junto con la autoridad GET para ejecutar transferencias de mensaje a archivo.

Desde IBM MQ 9.0.0 Fix Pack 2 y IBM MQ 9.0.4, Managed File Transfer se actualiza para restaurar la comprobación de comparaciones, eliminada anteriormente por el [APAR IT18213](#), del identificador de transferencia y el valor del atributo `groupId` dentro de la carga útil de XML de la solicitud de transferencia. Si estos dos identificadores son equivalentes, el agente de origen utiliza el identificador como una opción de coincidencia de identificador de mensaje, a diferencia de una opción de coincidencia de identificador de grupo, en el primer intento de MQGET que se realiza en la cola de entrada para llevar a cabo la transferencia de mensaje a archivo.

## Configuración de un agente para realizar transferencias de mensaje a archivo

De forma predeterminada, los agentes no pueden realizar transferencias de mensaje a archivo o archivo a mensaje. Para habilitar esta función, deberá establecer la propiedad del agente `enableQueueInputOutput` a `true`.

### Acerca de esta tarea

Si intenta realizar una transferencia de mensaje a archivo desde un agente de origen que no tiene la propiedad `enableQueueInputOutput` establecida en `true`, la transferencia falla. El mensaje de registro de transferencia publicado en el gestor de cola de coordinación contiene el siguiente mensaje:

```
BFGI00197E: An attempt to read from a queue was rejected by the source agent.  
The agent must have enableQueueInputOutput=true set in the agent.properties file  
to support transferring from a queue.
```

Para que el agente pueda escribir y leer en las colas, siga los siguientes pasos:

### Procedimiento

1. Detenga el agente de origen utilizando el mandato **fteStopAgent**.
2. Edite el archivo `agent.properties` para incluir la línea `enableQueueInputOutput=true`.  
El archivo `agent.properties` se encuentra en el directorio `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/source_agent_name`.
3. Inicie el agente de origen utilizando el mandato **fteStartAgent**.

### Ejemplo: Transferencia desde una cola a un único archivo

Puede especificar una cola de IBM MQ como origen de una transferencia de archivo utilizando el parámetro **-sq** con el mandato **fteCreateTransfer**.

### Acerca de esta tarea

Los datos de origen están contenidos en tres mensajes en la cola `START_QUEUE`. Esta cola debe estar en el gestor de colas del agente de origen, `QM_NEPTUNE`.

### Procedimiento

Escriba el siguiente mandato:

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE
```



El delimitador de texto se añade al principio de los datos de cada uno de los cuatro mensajes en START\_QUEUE mediante el agente de origen AGENT\_NEPTUNE. Estos datos se escriben en el archivo de destino, /out/output.txt.

## Ejemplo: Inserción de un delimitador binario después de los datos de cada mensaje

Cuando haga transferencias en modalidad binaria de una cola de origen a un archivo, puede especificar que se inserte un delimitador binario después de los datos de mensajes individuales utilizando los parámetros **-sq**, **-sqdb** y **-sqdp** con el mandato **fteCreateTransfer**.

### Acerca de esta tarea

En este ejemplo, existen tres mensajes en la cola START\_QUEUE. Esta cola está en el gestor de cola del agente de origen, QM\_NEPTUNE. El delimitador binario que insertar después de los datos de cada mensaje debe expresarse como una lista separada por comas de bytes hexadecimales, por ejemplo: x34,xE7,xAE.

### Procedimiento

Escriba el siguiente mandato:

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_VENUS -df /out/binary.file  
-sqdp postfix -sqdb x34,xE7,xAE -sq START_QUEUE
```

El agente de origen, AGENT\_NEPTUNE, añade el delimitador binario se añade a los datos de cada uno de los tres mensajes en START\_QUEUE. Estos datos se escriben en el archivo de destino, /out/binary.file.

## Supervisión de una cola y utilización de sustitución de variables

Puede supervisar una cola y transferir mensajes de la cola supervisada a un archivo utilizando el mandato **fteCreateMonitor**. El valor de cualquier propiedad de mensaje de IBM MQ en el primer mensaje que se va a leer de la cola supervisada se puede sustituir en la definición XML de la tarea y se utiliza para definir el comportamiento de la transferencia.

### Acerca de esta tarea

En este ejemplo, el agente de origen se denomina AGENT\_VENUS, que se conecta a QM\_VENUS. La cola que AGENT\_VENUS supervisa se denomina START\_QUEUE y se encuentra en QM\_VENUS. El agente sondea la cola cada 30 minutos.

Cuando un grupo completo de mensajes se graba en la cola, la tarea de supervisor envía el grupo de mensajes a un archivo en uno de los muchos agentes de destino, que se conectan todos ellos al gestor de colas QM\_MARS. El nombre del archivo al que se transfiere el grupo de mensajes se define mediante la propiedad de mensaje de IBM MQ `usr.fileName` en el primer mensaje del grupo. El nombre del agente al que se envía el grupo de mensajes se define mediante la propiedad de mensaje de IBM MQ `usr.toAgent` en el primer mensaje del grupo. Si la cabecera `usr.toAgent` no está establecida, el valor predeterminado que se va a utilizar para el agente de destino es AGENT\_MAGENTA.

Cuando especifica `useGroups="true"`, si no especifica también `groupId="${GROUPID}"`, la transferencia sólo toma el primer mensaje de la cola. Por ejemplo, si está utilizando la sustitución de variables para generar el nombre de archivo (`fileName`), es posible que el contenido de `a.txt` no sea correcto. Esto se debe a que el nombre de archivo (`fileName`) lo genera el supervisor, pero la transferencia obtiene realmente un mensaje que no es el que debe generar el archivo denominado `fileName`.



## Procedimiento

1. Cree el XML de tarea que define la tarea que el supervisor lleva a cabo cuando se desencadena.

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_VENUS" QMgr="QM_VENUS" />
    <destinationAgent agent="toAgent" QMgr="QM_MARS" />
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <queue useGroups="true" groupId="{GROUPID}">START_QUEUE</queue>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/{fileName}.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Las variables que se sustituyen por los valores de las cabeceras de mensajes de IBM MQ se resaltan en **negrita**. Este XML de tarea se guarda en el archivo /home/USER1/task.xml

2. Cree un supervisor de recursos para supervisar la cola START\_QUEUE.

Someta el siguiente mandato:

```
fteCreateMonitor -ma AGENT_VENUS -mm QM_VENUS -mq START_QUEUE
                 -mn myMonitor -mt /home/USER1/task.xml
                 -tr completeGroups -pi 30 -pu minutes -dv toAgent=AGENT_MAGENTA
```

3. Un usuario o un programa graba un grupo de mensajes en la cola START\_QUEUE.

El primer mensaje de este grupo cuenta con las siguientes propiedades de mensajes de IBM MQ establecidas:

```
usr.fileName=larmer
usr.toAgent=AGENT_VIOLET
```

4. El supervisor se desencadena cuando se ha grabado el grupo completo. El agente sustituye las propiedades de mensajes de IBM MQ en el XML de tarea.

El resultado es la transformación del XML de tarea en:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_VENUS" QMgr="QM_VENUS" />
    <destinationAgent agent="AGENT_VIOLET" QMgr="QM_MARS" />
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <queue useGroups="true" groupId="{GROUPID}">START_QUEUE</queue>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/larmer.rpt</file>
        </destination>
      </item>
    </transferSet>
```

```
</managedTransfer>  
</request>
```

## Resultados

Se realiza la transferencia definida por el XML de tarea. El grupo completo de mensajes leídos desde START\_QUEUE por AGENT\_VENUS se escribe en un archivo denominado /reports/larmer.rpt en el sistema donde AGENT\_VIOLET está en ejecución.

## Qué hacer a continuación

### Transferir cada mensaje a un archivo distinto

Si desea supervisar una cola y hacer que cada mensaje se transfiera a un archivo independiente, puede utilizar una técnica similar a la que se ha descrito anteriormente en este tema.

1. Cree el supervisor tal como se ha descrito anteriormente, especificando el parámetro **-tr completeGroups** en el mandato **fteCreateMonitor**.
2. En el XML de la tarea, especifique lo siguiente:

```
<queue useGroups="true" groupId="{GROUPLD}">START_QUEUE</queue>
```

Sin embargo, cuando coloque los mensajes en la cola de origen, no los coloque en un grupo de IBM MQ. Añada propiedades de mensaje de IBM MQ a cada mensaje. Por ejemplo, especifique la propiedad `usr.filename` con un valor de nombre de archivo exclusivo para cada mensaje. Esto hace que el Managed File Transfer Agent trate cada mensaje en la cola de origen como un grupo aparte.

## Ejemplo: Error al realizar una transferencia de un mensaje a un archivo utilizando las propiedades de mensaje de IBM MQ

Puede hacer que una transferencia de un mensaje a un archivo no se realice correctamente estableciendo la propiedad de mensaje `usr.UserReturnCode` IBM MQ en un valor distinto de cero. También puede especificar información suplementaria sobre la razón del error estableciendo la propiedad del mensaje `usr.UserSupplement` IBM MQ.

## Acerca de esta tarea

En este ejemplo, se está realizando una transferencia entre la cola INPUT\_QUEUE y el archivo /home/user/output.file.

Un usuario crea mensajes y los coloca en la cola INPUT\_QUEUE. El agente de origen consume mensajes de la cola INPUT\_QUEUE y envía los datos de la transferencia al agente de destino. El agente de destino está escribiendo estos datos en el archivo /home/user/output.file.

El usuario que graba mensajes en la cola INPUT\_QUEUE desea detener la transferencia que está en curso y suprimir los datos que ya se han grabado en el archivo de destino.

## Procedimiento

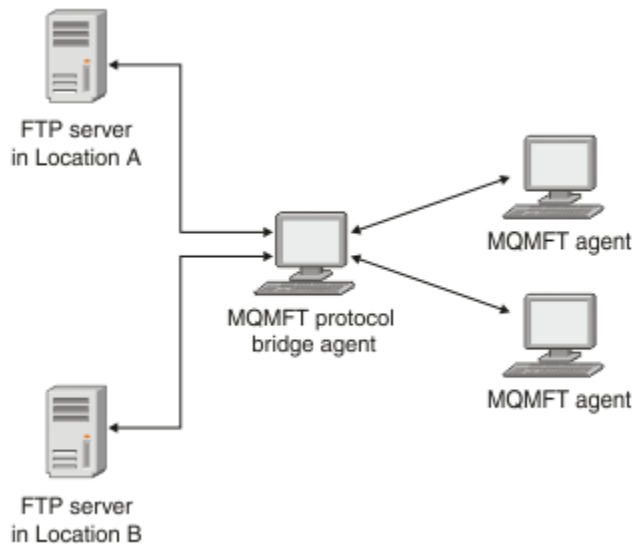
1. El usuario escribe un mensaje en la cola INPUT\_QUEUE que tiene establecidas las siguientes propiedades de mensajes de IBM MQ:

```
usr.UserReturnCode=1  
usr.UserSupplement="Cancelling transfer - sent wrong data."
```

2. El agente de origen lee las propiedades de mensajes de IBM MQ y detiene el proceso de mensajes de la cola. El agente de destino suprime los datos de archivo que se han grabado en el directorio de destino.
3. El agente de origen envía un mensaje de registro de transferencia al gestor de colas de coordinación que notifica la anomalía de la transferencia.



que permiten acceder a los archivos basándose sólo en el directorio actual no están soportados por el agente de puente de protocolo.



En el diagrama se muestran dos servidores FTP, en ubicaciones distintas. Los servidores FTP se utilizan para intercambiar archivos con los agentes de Managed File Transfer. El agente de puente de protocolo está entre los servidores FTP y el resto de la red de MFT y está configurado para comunicarse con ambos servidores FTP.

Asegúrese de tener otro agente en la red de MFT, además del agente de puente de protocolo. El agente de puente de protocolo es un puente sólo al servidor FTP, FTPS o SFTP y no graba archivos transferidos en el disco local. Si desea transferir archivos a o desde el servidor FTP, FTPS o SFTP, debe utilizar el agente de puente de protocolo como el destino u origen de la transferencia de archivos (en representación del servidor FTP, FTPS o SFTP) y otro agente estándar como el origen o destino correspondiente.

Cuando transfiera archivos utilizando el puente de protocolo, el puente debe tener permiso para leer el directorio de origen o de destino que contiene los archivos que desea transferir. Por ejemplo, si desea transferir archivos del directorio `/home/fte/bridge` que tiene solo permisos de ejecución (`d -- x -- x -- x`), las transferencias que intente desde este directorio fallan con el siguiente mensaje de error:

```
BFGBR0032E: Attempt to read filename from the protocol file server
has failed with server error 550. Failed to open file.
```

## Configuración de un agente de puente de protocolo

Un agente de puente de protocolo es como un agente de MFT estándar. Un agente de puente de protocolo se crea mediante el mandato **fteCreateBridgeAgent**. Puede configurar un agente de puente de protocolo utilizando el archivo `ProtocolBridgeProperties.xml`, que se describe en [Formato de archivo de propiedades de puente de protocolo](#). Si está utilizando una versión anterior, configure el agente utilizando las propiedades de puente de protocolo específicas que se describen en [Propiedades avanzadas de agente](#). Para todas las versiones, también puede configurar una correlación de credenciales como se describe en [“Correlación de credenciales para un servidor de archivos”](#) en la [página 292](#). Después de que haya configurado un agente de puente de protocolo para un servidor de archivos de protocolo específico, podrá utilizar ese agente con esa finalidad únicamente.

## Recuperación del puente de protocolo

Si el agente de puente de protocolo no puede conectarse al servidor de archivos porque el servidor de archivos está no disponible, todas las peticiones de transferencia de archivos se ponen en cola hasta que el servidor de archivos queda disponible. Si el agente de puente de protocolo no puede conectarse al servidor de archivos porque el agente está utilizando las credenciales incorrectas, la transferencia fallará

y el mensaje del registro de transferencias reflejará este error. Si el agente de puente de protocolo finaliza por cualquier motivo, todas las transferencias de archivos solicitadas se retienen y continúan cuando se reinicia el puente de protocolo.

Durante la transferencia de archivos, los archivos suelen grabarse como archivos temporales en el destino y, a continuación, se renombran cuando termina la transferencia. Sin embargo, si el destino de la transferencia es un servidor de archivos de protocolo que está configurado como de grabación limitada (los usuarios pueden subir archivos al servidor de archivos de protocolo pero no pueden cambiar esos archivos subidos de ninguna manera; en realidad los usuarios sólo pueden grabar una vez), los archivos transferidos se graban directamente en el destino. Esto significa que si se produce un problema durante la transferencia, los archivos grabados parcialmente permanecen en el servidor de archivos de protocolo de destino y Managed File Transfer no puede suprimirlos ni editarlos. En esta situación, la transferencia no se realiza satisfactoriamente.

## Definición de propiedades para servidores de archivos de protocolo utilizando el archivo `ProtocolBridgeProperties.xml`

Defina las propiedades de uno o más servidores de archivos de protocolo a los que desea transferir archivos y mediante el archivo `ProtocolBridgeProperties.xml`, que Managed File Transfer proporciona en el directorio de configuración del agente.

### Acerca de esta tarea

El mandato `fteCreateBridgeAgent` crea el archivo `ProtocolBridgeProperties.xml` en el directorio de configuración del agente `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name`. El mandato también crea una entrada en el archivo para el servidor de archivos de protocolo predeterminado, si se especificó un valor predeterminado cuando se ejecutó el mandato.

El mensaje BFGCL0392I proporciona la ubicación del archivo `ProtocolBridgeProperties.xml`.

```
<?xml version="1.0" encoding="IBM-1047"?>
<!--
This ProtocolBridgeProperties.xml file determines the protocol servers that will be accessed by
the
MQMFT protocol bridge agent.

Each protocol server is defined using either a <tns:ftpServer>, <tns:ftpsServer>, or
<tns:sftpServer>
element - depending on the protocol used to communicate with the server. When the protocol
bridge agent participates in a managed file transfer it will determine which server to used
based on
the prefix (if any) present on the file path. For example a file path of 'server1:/home/user/
file.txt' would
be interpreted as a request to transfer /home/user/file.txt using 'server1'. The server name
is compared
to the 'name' attribute of each <tns:ftpServer>, <tns:ftpsServer> or <tns:sftpServer> element
in this
XML document and the first match is used to determine which protocol server the protocol bridge
agent will connect to. If no match is found then the managed file transfer operation will fail.

If a file path is not prefixed with a server name, for example '/home/user/file.txt' then this
XML
document can specify a default server to use for the managed file transfer. To specify a
default server use the <tns:defaultServer> element as the first element inside the
<tns:serverProperties>
element. The default server will be used whenever the protocol bridge agent participates in
a managed file transfer for file names which do not specify a prefix.

An optional <tns:limits> element can be specified within each server definition. This element
contains
attributes that govern the amount of resources used by each defined server.

An optional <tns:credentialsFile> element can be specified within each serverProperties
definition. This
element contains a path to a file containing credentials to be used when connecting to defined
servers.

An example ProtocolBridgeProperties.xml file is as follows:
```

```

<?xml version="1.0" encoding="UTF-8"?>
<tns:serverProperties xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeProperties"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeProperties
  ProtocolBridgeProperties.xsd">
  <tns:credentialsFile path="$HOME/ProtocolBridgeCredentials.xml" />
  <tns:defaultServer name="myFTPserver" />
  <tns:ftpServer name="myFTPserver" host="windows.hursley.ibm.com" port="1234"
  platform="windows"
  timeZone="Europe/London" locale="en_GB" fileEncoding="UTF-8"
  listFormat="unix" limitedWrite="false">
    <tns:limits maxListFileNames="100" maxListDirectoryLevels="999999999"
      maxReconnectRetry="2" reconnectWaitPeriod="10"
      maxSessions="60" socketTimeout="30" />
  </tns:ftpServer>
  <tns:ftpsServer name="myFTPSserver" host="unix.hursley.ibm.com" platform="unix"
  timeZone="Europe/London" locale="en_GB" fileEncoding="UTF8"
  listFormat="unix" limitedWrite="false" ftpsType="explicit"
  trustStore="C:\FTE\keystores\myFTPSserver\FTPSKeyStore.jks"
  trustStorePassword="password">
    <tns:limits maxReconnectRetry="10" connectionTimeout="10"/>
  </tns:ftpsServer>
  <tns:sftpServer name="mySFTPSserver" host="windows.hursley.ibm.com" platform="windows"
  timeZone="Europe/London" locale="en_GB" fileEncoding="UTF-8"
  limitedWrite="false">
    <tns:limits connectionTimeout="60"/>
  </tns:sftpServer>
</tns:serverProperties>

```

This example shows the outermost `<tns:serverProperties>` element which must exist for the document to be valid, an optional `<tns:defaultServer>` element, as well as definitions for an FTP, FTPS and SFTP server.

The attributes of the `<tns:ftpServer>`, `<tns:ftpsServer>` and `<tns:sftpServer>` elements determine the characteristics of the connection established to the server. These attributes correspond to the command line parameters for the 'fteCreateBridgeAgent' command.

The following attributes are valid for all of the `<tns:ftpServer>`, `<tns:ftpsServer>` and `<tns:sftpServer>` elements: name, host, port, platform, fileEncoding, limitedWrite and controlEncoding.

The following attributes are valid for the `<tns:ftpServer>` and `<tns:ftpsServer>` elements: timeZone, locale, listFormat, listFileRecentDateFormat, listFileOldDateFormat, and monthShortNames.

The following attributes are valid for the `<tns:ftpServer>` element only: passiveMode

The following attributes are valid for the `<tns:ftpsServer>` element only: ftpsType, trustStore, trustStorePassword, trustStoreType, keyStore, keyStorePassword, keyStoreType, ccc, protFirst, auth, and connectTimeout.

The following attributes are valid for the `<tns:limits>` element within all of the `<tns:ftpServer>`, `<tns:ftpsServer>` and `<tns:sftpServer>` elements: maxListFileNames, maxListDirectoryLevels, maxReconnectRetry, reconnectWaitPeriod, maxSessions and socketTimeout

```

-->
<tns:serverProperties xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeProperties"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeProperties
  ProtocolBridgeProperties.xsd">

```

```

  <!-- By default the location of the credentials file is in the home directory of the user
  that started the -->
  <!-- protocol bridge agent. If you wish to specify a different location use the

```

```

credentialsFile element to -->
  <!-- do this. For
example:
  <!-- <tns:credentialsFile path="/test/
ProtocolBridgeCredentials.xml"/>
-->

  <tns:defaultServer name="WINMVSCA.HURSLEY.IBM.COM" />
  <tns:ftpServer name="WINMVSCA.HURSLEY.IBM.COM" host="WINMVSCA.HURSLEY.IBM.COM"
platform="UNIX"
      timeZone="Europe/London" locale="en-GB" fileEncoding="US-ASCII"
      listFormat="unix" limitedWrite="false" />

  <!-- Define servers here -->
</tns:serverProperties>

```

El mandato puede generar el mensaje BFGCL0532I siguiente:

Para que este agente funcione, debe crearse manualmente un archivo de credenciales adicionales. De forma predeterminada, el nombre de este archivo es ProtocolBridgeCredentials.xml y se encuentra en el directorio de inicio del usuario que inicia el agente. Por ejemplo, si este usuario ha iniciado el agente, la ubicación será: \$HOME/ProtocolBridgeCredentials.xml

Si utiliza un archivo de credenciales:

1. Consulte el texto siguiente para obtener más información acerca de cómo crear uno.
2. El archivo de credenciales debe estar en un directorio con permisos restringidos. Por ejemplo, otros usuarios no deben tener acceso de lectura.
3. Especifique la ubicación del directorio del archivo de credenciales en la variable de entorno \$HOME para el ID de usuario del agente iniciado, o edite el archivo ProtocolBridgeProperties.xml y especifique la ubicación en:

```
<tns:credentialsFile path="/test/ProtocolBridgeCredentials.xml"/>
```

Si desea añadir servidores de protocolo no predeterminados adicionales, edite este archivo para definir sus propiedades. Este ejemplo añade un servidor FTP adicional.

**Nota:** El agente de puente de protocolo no da soporte al bloqueo de archivo. Esto se debe a que Managed File Transfer no da soporte al mecanismo de bloqueo de archivo en un servidor de archivos.

## Procedimiento

1. Defina un servidor de archivos de protocolo insertando las líneas siguientes en el archivo como elemento hijo de <tns:serverProperties>:

```

<tns:ftpServer name="myserver" host="myhost.hursley.ibm.com" port="1234"
platform="windows"
      timeZone="Europe/London" locale="en-GB" fileEncoding="UTF-8"
      listFormat="unix" limitedWrite="false" >
<tns:limits maxListFileNames="10" maxListDirectoryLevels="500"/>

```

2. A continuación, cambie el valor de los atributos:

- name es el nombre de su servidor de archivos de protocolo
- host es el nombre de host o dirección IP del servidor de archivos de protocolo
- port es el número de puerto del servidor de archivos de protocolo
- platform es la plataforma en la que se ejecuta el servidor de archivos de protocolo
- timeZone es el huso horario en el que se ejecuta el servidor de archivos de protocolo
- locale es el idioma utilizado en el servidor de archivos de protocolo
- fileEncoding es la codificación de caracteres del servidor de archivos de protocolo
- listFormat es el formato de listado de archivos devuelto por el servidor de archivos de protocolo
- limitedWrite determina si se debe seguir la modalidad predeterminada al grabar en un servidor de archivos, que consiste en crear un archivo temporal y luego renombrar ese archivo cuando la transferencia se ha completado. Para un servidor de archivos que está configurado como de sólo grabación, el archivo se crea directamente con su nombre final. El valor de esta propiedad puede ser

true o false. El atributo `limitedWrite` y la propiedad de agente `doNotUseTempOutputFile` se usan juntos en el caso de tratarse de agentes de puente de protocolo. Si desea utilizar archivos temporales, no debe establecer el valor de `doNotUseTempOutputFile`, y debe establecer el valor de `limitedWrite` en false. Cualquier otra combinación de valores significa que los archivos temporales no se utilizarán.

- `maxListFileNames` es el número máximo de nombres que se recopilan al explorar un directorio en el servidor de archivos de protocolo para detectar nombres de archivo.
- `maxListDirectoryLevels` es el número máximo de niveles de directorio en los que buscar de forma recursiva al explorar un directorio en el servidor de archivos de protocolo para detectar nombres de archivo.

Si desea más detalles sobre estos atributos, incluyendo si son necesarios u opcionales y sus valores predeterminados, consulte [Formato de archivo de propiedades de puente de protocolo](#).

### Referencia relacionada

[Formato del archivo de propiedades de puente de protocolo](#)

[Expresiones regulares utilizadas por MFT](#)

## ProtocolBridgePropertiesExit2: buscar propiedades de servidor de archivos de protocolo

Si tiene un gran número de servidores de archivos de protocolo, puede implementar la interfaz `com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit2` para buscar propiedades de servidor de archivos de protocolo a las que se hace referencia en las transferencias. Puede implementar esta interfaz de preferencia para mantener un archivo `ProtocolBridgeProperties.xml`. Managed File Transfer proporciona una salida de usuario de ejemplo que busca propiedades de servidor de archivos de protocolo.

### *Configurar las salidas de usuario que buscan propiedades de puente de protocolo*

#### Acerca de esta tarea

Cualquier salida de usuario que busque propiedades de puente de protocolo debe implementar la interfaz `com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit2`. Si desea más información, consulte [Interfaz ProtocolBridgePropertiesExit.java](#).

Puede encadenar varias salidas de propiedades de servidor de protocolo de forma parecida a otras salidas de usuario. Las salidas se invocan en el orden en que se especifican utilizando la propiedad `protocolBridgePropertiesExitClasses` en el archivo de propiedades de agente. Los métodos `initialize` todos regresan por separado y si uno o más devuelve un valor de false, el agente no se inicia. El error se comunica en el registro de sucesos del agente.

Sólo se devuelve un resultado global para los métodos `getProtocolServerProperties` de todas las salidas. Si el método devuelve un objeto de propiedades como el código de resultado, este valor es el resultado devuelto y no se llama a los métodos `getProtocolServerProperties` de las salidas subsiguientes. Si el método devuelve un valor de null como el código de resultado, se llama al método `getProtocolServerProperties` de la salida siguiente. Si no hay ninguna salida subsiguiente, se devuelve el resultado null. El agente de puente de protocolo considera un código de resultado global de null como un error de búsqueda.

Se recomienda que utilice la interfaz `ProtocolBridgePropertiesExit2.java`, pero para obtener información sobre la interfaz `ProtocolBridgePropertiesExit.java`, consulte [“ProtocolBridgePropertiesExit: buscar propiedades de servidor de archivos de protocolo”](#) en la página 290.

Para ejecutar la salida, realice los pasos siguientes:

#### Procedimiento

1. Compile la salida de usuario de propiedades de servidor de protocolo.



2. Cree un archivo de archivado Java (JAR) que contenga la salida compilada y su estructura de paquetes.
3. Coloque el archivo JAR que contiene la clase de salida en el directorio `exits` del agente de puente de protocolo. Este directorio se encuentra en el directorio `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name`.
4. Edite el archivo de propiedades del agente de puente de protocolo para incluir la propiedad `protocolBridgePropertiesExitClasses`. Para el valor de esta propiedad, especifique una lista separada por comas de clases que implementan una salida de usuario de propiedades de servidor de puente de protocolo. Las clases de salida se invocan en el orden en que están especificadas en esta lista. Para obtener más información, consulte [El archivo MFT agent.properties](#).
5. Opcionalmente, puede especificar la propiedad `protocolBridgePropertiesConfiguration`. El valor que especifique para esta propiedad se pasa como una Serie al método `initialize()` de las clases de salida especificadas mediante `protocolBridgePropertiesExitClasses`. Para obtener más información, consulte [El archivo MFT agent.properties](#).

## Utilización de la salida de usuario de ejemplo

### Acerca de esta tarea

Se proporciona una salida de usuario de ejemplo que busca propiedades de puente de protocolo en el directorio `MQ_INSTALLATION_PATH/mqft/samples/protocolBridge` y en el tema [Ejemplo de salida de usuario de propiedades de puente de protocolo](#).

La salida `SamplePropertiesExit2.java` lee un archivo de propiedades que contiene propiedades para servidores de protocolo. El formato de cada entrada del archivo de propiedades es el siguiente:

```
serverName=type://host:port
```

La ubicación del archivo de propiedades se determina a partir de la propiedad de agente de puente de protocolo `protocolBridgePropertiesConfiguration`.

Para ejecutar la salida de usuario de ejemplo, realice los pasos siguientes:

### Procedimiento

1. Compile el archivo `SamplePropertiesExit2.java`.
2. Cree un archivo JAR que contenga la salida compilada y su estructura de paquetes.
3. Coloque el archivo JAR en el directorio `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent/exits`.
4. Edite el archivo `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name/agent.properties` para que contenga la línea:

```
protocolBridgePropertiesExitClasses=SamplePropertiesExit2
```

5. Cree un archivo de propiedades de puente de protocolo, por ejemplo `protocol_bridge_properties.properties`, en el directorio `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent`. Edite este archivo para incluir entradas en el formato:

```
serverName=type://host:port
```

6. Edite el archivo `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent/agent.properties` para que contenga la línea:

```
protocolBridgePropertiesConfiguration=MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent/protocol_bridge_properties.properties
```

Debe utilizar la vía de acceso absoluta al archivo `protocol_bridge_properties.properties`.

7. Inicie el agente de puente de protocolo utilizando el mandato **fteStartAgent**.

### Conceptos relacionados

[“El puente de protocolo” en la página 283](#)

El puente de protocolo permite que la red de Managed File Transfer (MFT) acceda a los archivos almacenados en un servidor de archivos fuera de la red de MFT, bien en su dominio local o bien en una ubicación remota. Este servidor de archivos puede utilizar los protocolos de red FTP, FTPS o SFTP. Cada servidor de archivos necesita al menos un agente dedicado. El agente dedicado se conoce como el agente de puente de protocolo. Un agente de puente puede interactuar con varios servidores de archivos.

### Referencia relacionada

[Interfaz ProtocolBridgePropertiesExit.java](#)

[Ejemplo de salida de usuario de propiedades de puente de protocolo](#)

[El archivo MFT agent.properties](#)

[fteCreateBridgeAgent \(crear y configurar un agente de puente de protocolo de MFT\)](#)

### ***ProtocolBridgePropertiesExit: buscar propiedades de servidor de archivos de protocolo***

Si tiene un gran número de servidores de archivos de protocolo, puede implementar la interfaz `com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit` para buscar propiedades de servidor de archivos de protocolo a las que se hace referencia en las transferencias. Puede implementar esta interfaz de preferencia para mantener un archivo `ProtocolBridgeProperties.xml`. Se recomienda utilizar la interfaz `ProtocolBridgePropertiesExit2.java`, pero la interfaz `ProtocolBridgePropertiesExit.java` también está soportada. Si tiene una implementación existente de la interfaz `ProtocolBridgePropertiesExit.java` de IBM WebSphere MQ File Transfer Edition, puede utilizarla en IBM WebSphere MQ 7.5 o posteriores. El nuevo método `getCredentialLocation` de `ProtocolBridgePropertiesExit2.java` utiliza la ubicación predeterminada del archivo `ProtocolBridgeCredentials.xml`, que es su directorio inicial.

### **Configurar las salidas de usuario que buscan propiedades de puente de protocolo**

Cualquier salida de usuario que busque propiedades de puente de protocolo debe implementar la interfaz `com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit`. Para obtener más información, consulte [“Interfaz ProtocolBridgePropertiesExit.java” en la página 291](#).

Puede encadenar varias salidas de propiedades de servidor de protocolo de forma parecida a otras salidas de usuario. Las salidas se invocan en el orden en que se especifican utilizando la propiedad `protocolBridgePropertiesExitClasses` en el archivo de propiedades de agente. Los métodos `initialize` todos regresan por separado y si uno o más devuelve un valor de `false`, el agente no se inicia. El error se comunica en el registro de sucesos del agente.

Sólo se devuelve un resultado global para los métodos `getProtocolServerProperties` de todas las salidas. Si el método devuelve un objeto de propiedades como el código de resultado, este valor es el resultado devuelto y no se llama a los métodos `getProtocolServerProperties` de las salidas subsiguientes. Si el método devuelve un valor de `null` como el código de resultado, se llama al método `getProtocolServerProperties` de la salida siguiente. Si no hay ninguna salida subsiguiente, se devuelve el resultado `null`. El agente de puente de protocolo considera un código de resultado global de `null` como un error de búsqueda.

Para ejecutar la salida, realice los pasos siguientes:

1. Compile la salida de usuario de propiedades de servidor de protocolo.
2. Cree un archivo de archivado Java (JAR) que contenga la salida compilada y su estructura de paquetes.
3. Coloque el archivo JAR que contiene la clase de salida en el directorio `exits` del agente de puente de protocolo. Este directorio se encuentra en el directorio `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name`.
4. Edite el archivo de propiedades del agente de puente de protocolo para incluir la propiedad `protocolBridgePropertiesExitClasses`. Para el valor de esta propiedad, especifique una lista

separada por comas de clases que implementan una salida de usuario de propiedades de servidor de puente de protocolo. Las clases de salida se invocan en el orden en que están especificadas en esta lista. Para obtener más información, consulte [El archivo MFT agent.properties](#).

5. Opcionalmente, puede especificar la propiedad `protocolBridgePropertiesConfiguration`. El valor que especifique para esta propiedad se pasa como una Serie al método `initialize()` de las clases de salida especificadas mediante `protocolBridgePropertiesExitClasses`. Para obtener más información, consulte [El archivo MFT agent.properties](#).

## Interfaz `ProtocolBridgePropertiesExit.java`

```
package com.ibm.wmqfte.exitroutine.api;

import java.util.Map;
import java.util.Properties;

/**
 * An interface that is implemented by classes that are to be invoked as part of
 * user exit routine processing. This interface defines methods that will be
 * invoked by a protocol bridge agent to look up properties for protocol servers
 * that are referenced in transfers.
 * <p>
 * There will be one instance of each implementation class for each protocol
 * bridge agent. The methods can be called from different threads so the methods
 * must be synchronised.
 */
public interface ProtocolBridgePropertiesExit {

    /**
     * Invoked once when a protocol bridge agent is started. It is intended to
     * initialize any resources that are required by the exit.
     *
     * @param bridgeProperties
     *     The values of properties defined for the protocol bridge.
     *     These values can only be read, they cannot be updated by the
     *     implementation.
     * @return {@code true} if the initialization is successful and {@code
     *     false} if unsuccessful. If {@code false} is returned from an exit
     *     the protocol bridge agent will not start.
     */
    public boolean initialize(final Map<String, String> bridgeProperties);

    /**
     * Obtains a set of properties for the specified protocol server name.
     * <p>
     * The returned {@link Properties} must contain entries with key names
     * corresponding to the constants defined in
     * {@link ProtocolServerPropertyConstants} and in particular must include an
     * entry for all appropriate constants described as required.
     *
     * @param protocolServerName
     *     The name of the protocol server whose properties are to be
     *     returned. If a null or a blank value is specified, properties
     *     for the default protocol server are to be returned.
     * @return The {@link Properties} for the specified protocol server, or null
     *     if the server cannot be found.
     */
    public Properties getProtocolServerProperties(
        final String protocolServerName);

    /**
     * Invoked once when a protocol bridge agent is shut down. It is intended to
     * release any resources that were allocated by the exit.
     *
     * @param bridgeProperties
     *     The values of properties defined for the protocol bridge.
     *     These values can only be read, they cannot be updated by the
     *     implementation.
     */
    public void shutdown(final Map<String, String> bridgeProperties);
}
```

## Correlación de credenciales para un servidor de archivos

Correlacione credenciales de usuario de Managed File Transfer con credenciales de usuario del servidor de archivos, utilizando la función de correlación de credenciales predeterminada del agente de puente de protocolo o escribiendo su propia salida de usuario. Managed File Transfer proporciona una salida de usuario de ejemplo que realiza la correlación de credenciales de usuario.

### **Correlación de credenciales para un servidor de archivos utilizando el archivo *ProtocolBridgeCredentials.xml***

Correlacione credenciales de usuario de Managed File Transfer con credenciales de usuario del servidor de archivos, utilizando la función de correlación de credenciales predeterminada del agente de puente de protocolo. Managed File Transfer proporciona un archivo XML que puede editar para incluir información sobre las credenciales.

### **Acerca de esta tarea**

El usuario debe crear manualmente el archivo `ProtocolBridgeCredentials.xml`. De forma predeterminada, la ubicación de este archivo es el directorio inicial del usuario que ha iniciado el agente de puente de protocolo, pero puede almacenarse en cualquier parte del sistema de archivos al que pueda acceder el agente. Para especificar una ubicación diferente, añada el elemento `<credentialsFile>` al archivo `ProtocolBridgeProperties.xml`. Por ejemplo,

```
<tns:credentialsFile path="/example/path/to/ProtocolBridgeCredentials.xml"/>
```

Antes de poder utilizar un agente de puente de protocolo, configure la correlación de credenciales editando este archivo para incluir información de host, de usuario y de credenciales. Si desea más información y ejemplos, consulte [Formato de archivo de credenciales de puente de protocolo](#).

**z/OS** Al crear el archivo `ProtocolBridgeCredentials.xml` en una plataforma z/OS utilizando IBM WebSphere MQ 7.5 o anterior, debe establecer un código de archivo antes de editar el archivo. Ejecute el mandato siguiente para marcar el archivo como archivo con contenido ASCII:

```
chtag -t -c IS08859-1 ProtocolBridgeCredentials.xml
```

**Nota:** **z/OS** En z/OS, puede almacenar el archivo de credenciales de puente de protocolo en un conjunto de datos, donde el usuario puede especificar el nombre del archivo `.xml`.

## Procedimiento

1. Edite la línea `<tns:server name="server name">` para cambiar el valor del atributo de nombre por el nombre de servidor en el archivo `ProtocolBridgeProperties.xml`.

Los agentes de puente de protocolo que se crean para IBM WebSphere MQ File Transfer Edition 7.0.4 y anteriores no tienen un archivo `ProtocolBridgeProperties.xml` (o salidas de usuario relacionadas), por lo que para IBM WebSphere MQ File Transfer Edition 7.0.4 Fix Pack 1 y posterior se asigna automáticamente el nombre de servidor al nombre de host del servidor. Por consiguiente, si utiliza un archivo `ProtocolBridgeCredentials.xml` actualizado con entradas `<server>`, coincidirá un nombre que corresponda al nombre de host del servidor.

Puede utilizar el atributo de patrón (`pattern`) para especificar que ha utilizado un nombre de servidor que contiene comodines o expresiones regulares. Por ejemplo,

```
<tns:server name="serverA*" pattern="wildcard">
```

2. Inserte el ID de usuario y la información de credenciales en el archivo como elementos hijo de `<tns:server>`.

Puede insertar uno o varios de los siguientes elementos en el archivo:

- Si el servidor de archivos de protocolo es un servidor FTP, FTPS o SFTP, puede utilizar contraseñas para autenticar el usuario que solicita la transferencia. Inserte las siguientes líneas en el archivo:

```
<tns:user name="FTE User ID"
  serverUserId="Server User ID"
  serverPassword="Server Password">
</tns:user>
```

Cambie el valor de los atributos.

- name es una expresión regular Java para comparar el ID de usuario MQMD asociado con la solicitud de transferencia de MFT
- serverUserId es el valor que se pasa al servidor de archivos de protocolo como ID de usuario de inicio de sesión. Si el atributo serverUserId no se especifica, se utiliza en su lugar el ID de usuario MQMD asociado a la solicitud de transferencia de MFT.
- serverPassword es la contraseña que se asocia con serverUserId.

El atributo name puede contener una expresión regular Java. El correlacionador de credenciales intenta emparejar el ID de usuario MQMD de la solicitud de transferencia de MFT con esta expresión regular. El agente de puente de protocolo intenta emparejar el ID de usuario MQMD con la expresión regular del atributo name de los elementos <tns:user> en el orden en que los elementos existen en el archivo. Cuando se encuentra una coincidencia, el agente de puente de protocolo no busca más coincidencias. Si se encuentra una coincidencia, los valores serverUserId y serverPassword correspondientes se pasan al servidor de archivos de protocolo como el ID de usuario y la contraseña de inicio de sesión. Las coincidencias de ID de usuario MQMD distinguen entre mayúsculas y minúsculas.


- Si el servidor de archivos de protocolo es un servidor SFTP, puede utilizar claves públicas y privadas para autenticar el usuario que solicita la transferencia. Inserte las líneas siguientes en el archivo y cambie el valor de los atributos. El elemento <tns:user> puede contener uno o varios elementos <tns:privateKey>.

```
<tns:user name="FTE User ID"
  serverUserId="Server User ID"
  hostKey="Host Key">
  <tns:privateKey associationName="association"
    keyPassword="Private key password">
    Private key file text
  </tns:privateKey>
</tns:user>
```

- name es una expresión regular Java para comparar el ID de usuario MQMD asociado con la solicitud de transferencia de MFT
- serverUserId es el valor que se pasa al servidor de archivos de protocolo como ID de usuario de inicio de sesión. Si el atributo serverUserId no se especifica, se utiliza en su lugar el ID de usuario MQMD asociado a la solicitud de transferencia de MFT.
- hostKey es la clave esperada que se devuelve del servidor al iniciar la sesión
- key es la clave privada de serverUserId
- keyPassword es la contraseña de la clave para generar claves públicas
- associationName es un valor que se utiliza para la identificación para el rastreo y registro cronológico

El atributo name puede contener una expresión regular Java. El correlacionador de credenciales intenta emparejar el ID de usuario MQMD de la solicitud de transferencia de MFT con esta expresión regular. El agente de puente de protocolo intenta emparejar el ID de usuario MQMD con la expresión regular del atributo name de los elementos <tns:user> en el orden en que los elementos existen en el archivo. Cuando se encuentra una coincidencia, el agente de puente de protocolo no busca más coincidencias. Si se encuentra una coincidencia, se utilizan los valores de serverUserId y key correspondientes para autenticar el usuario MFT con el servidor de archivos de protocolo. Las coincidencias de ID de usuario MQMD distinguen entre mayúsculas y minúsculas.

Para obtener más información sobre la utilización de claves privadas con un agente de puente de protocolo, consulte [“Ejemplo: Cómo configurar un agente de puente de protocolo para utilizar credenciales de clave privada con un servidor SFTP UNIX”](#) en la página 296.

**Nota:**  z/OS

Cuando la solicitud de transferencia se graba en la cola de mandatos, es posible que el ID de usuario MQMD se convierta a mayúsculas si la cola de mandatos del agente de origen se encuentra en un sistema z/OS o IBM i. En consecuencia, el ID de usuario de MQMD del mismo usuario de origen puede llegar a la salida de credenciales con las letras originales o convertidas a mayúsculas, en función del agente de origen especificado en la solicitud de transferencia. La salida de correlación de credenciales predeterminada realiza una comprobación de coincidencia de mayúsculas/minúsculas en el ID de usuario de MQMD suministrado, que podría tener que habilitar en el archivo de correlación

### ***Correlacionar credenciales para un servidor de archivos utilizando clases de salida***

Si no desea utilizar la función de correlación de credenciales predeterminada del agente de puente de protocolo, puede correlacionar credenciales de usuario de Managed File Transfer con credenciales de usuario del servidor de archivos escribiendo su propia salida de usuario. Managed File Transfer proporciona una salida de usuario de ejemplo que realiza la correlación de credenciales de usuario. Si configura salidas de usuario de correlación de credenciales, éstas sustituyen a la función de correlación de credenciales predeterminada.

*Configuración de salidas de usuario de credenciales de puente de protocolo*

### **Acerca de esta tarea**

Una salida de usuario para correlacionar credenciales de puente de protocolo debe implementar una de las siguientes interfaces:

- `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit`, que permite a un agente de puente de protocolo transferir archivos a y desde un servidor de archivos de protocolo predeterminado
- `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit2`, que le permite transferir archivos a y desde varios puntos finales.

La interfaz `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit2` contiene la misma función que `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit` y también incluye una función ampliada. Si desea más información, consulte [Interfaz ProtocolBridgeCredentialExit.java](#) e [Interfaz ProtocolBridgeCredentialExit2.java](#).

Las salidas de credenciales pueden encadenarse juntas de forma parecida a otras salidas de usuario. Las salidas se invocan en el orden en que se especifican utilizando la propiedad `protocolBridgeCredentialConfiguration` en el archivo de propiedades de agente. Los métodos `initialize` todos regresan por separado y si uno o más devuelve un valor de `false`, el agente no se inicia. El error se comunica en el registro de sucesos del agente.

Sólo se devuelve un resultado global para los métodos `mapMQUserId` de todas las salidas, como se indica a continuación:

- Si el método devuelve un valor de `USER_SUCCESSFULLY_MAPPED` o `USER_DENIED_ACCESS` como el código de resultado, este valor es el resultado devuelto y no se llama a los métodos `mapMQUserId` de las salidas subsiguientes.
- Si el método devuelve un valor `NO_MAPPING_FOUND` como el código de resultado, se llama al método `mqMQUserId` de la salida siguiente.
- Si no existe ninguna salida subsiguiente, se devuelve el resultado `NO_MAPPING_FOUND`.
- El agente de puente considera un código de resultado global de `USER_DENIED_ACCESS` o `NO_MAPPING_FOUND` como una anomalía en la transferencia.

Para ejecutar la salida, realice los pasos siguientes:

## Procedimiento

1. Compile la salida de usuario de credenciales de puente de protocolo.
2. Cree un archivo de archivado Java (JAR) que contenga la salida compilada y su estructura de paquetes.
3. Coloque el archivo JAR que contiene la clase de salida en el directorio `exits` del agente de puente. El directorio está en el directorio `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name`.
4. Edite el archivo de propiedades del agente de puente de protocolo para incluir la propiedad `protocolBridgeCredentialExitClasses`. Para el valor de esta propiedad, especifique una lista de clases, separadas por comas, que implementan una rutina de salida de credenciales de puente de protocolo. Las clases de salida se invocan en el orden en que están especificadas en esta lista. Para obtener más información, consulte [El archivo MFT agent.properties](#).
5. Edite el archivo de propiedades del agente de puente de protocolo para incluir:

```
exitClassPath=IBM MQ
installation_directory\mqft\config\configuration_queue_manager\agents\protocol_bridge_agent_n
ame\exits\SampleCredentialExit.jar
```

El archivo `agent.properties` para un agente se encuentra en el directorio `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/bridge_agent_name`.

Si cambia el archivo `agent.properties`, debe reiniciar el agente para recoger los cambios.

6. Puede especificar opcionalmente la propiedad `protocolBridgeCredentialConfiguration`. El valor que especifique para esta propiedad se pasa como un objeto `String` al método `initialize()` de las clases de salida especificadas mediante `protocolBridgeCredentialExitClasses`. Para obtener más información, consulte [El archivo MFT agent.properties](#).
7. Inicie el agente de puente de protocolo con el mandato **fteStartAgent**.

*Utilización de la salida de usuario de ejemplo*

## Acerca de esta tarea

Se proporciona una salida de credencial de puente de protocolo de ejemplo en el directorio `MQ_INSTALLATION_PATH/mqft/samples/protocolBridge` y en el tema [Ejemplo de salida de usuario de credenciales de puente de protocolo](#). Este ejemplo se basa en la interfaz `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit`.

La salida `SampleCredentialExit.java` lee un archivo de propiedades que correlaciona los ID de usuario MQMD asociados con las solicitudes de transferencia a los ID de usuario del servidor y a las contraseñas del servidor. La ubicación del archivo de propiedades se extrae de la propiedad de agente de puente de protocolo `protocolBridgeCredentialConfiguration`.

Para ejecutar la salida de usuario de ejemplo, realice los pasos siguientes:

## Procedimiento

1. Compile el archivo `SampleCredentialExit.java`.
2. Cree un archivo JAR que contenga la salida compilada y su estructura de paquetes.
3. Coloque el archivo JAR en el directorio `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name/exits`.
4. Edite el archivo `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name/agent.properties` para que contenga la línea:

```
protocolBridgeCredentialExitClasses=SampleCredentialExit
```

5. Edite el archivo de propiedades del agente de puente de protocolo para incluir:

```
exitClassPath=IBM MQ
installation_directory\mqft\config\configuration_queue_manager\agents\protocol_bridge_agent_name\exits\SampleCredentialExit.jar
```

El archivo `agent.properties` para un agente se encuentra en el directorio `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name`.

Si cambia el archivo `agent.properties`, debe reiniciar el agente para recoger los cambios.

6. Cree un archivo de propiedades de credenciales (`credentials.properties`) en el directorio `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent` y edítelo para incluir entradas en el formato:

```
mqUserId=serverUserId,serverPassword
```

7. Edite el archivo `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name/agent.properties` para que contenga la línea:

```
protocolBridgeCredentialConfiguration=MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name/credentials.properties
```

Debe utilizar la vía de acceso absoluta al archivo `credentials.properties`.

8. Inicie el agente de puente de protocolo utilizando el mandato **fteStartAgent**.

## Ejemplo: Cómo configurar un agente de puente de protocolo para utilizar credenciales de clave privada con un servidor SFTP UNIX

Este ejemplo muestra cómo puede generar y configurar el archivo `ProtocolBridgeCredentials.xml`. Este ejemplo es un ejemplo típico y los detalles pueden variar en función de la plataforma, pero los principios siguen siendo los mismos.

### Acerca de esta tarea

#### Procedimiento

1. Genere una clave pública y privada que se utilizará para autenticarse con el servidor SFTP.

Por ejemplo, en un sistema host Linux, puede utilizar la herramienta **ssh-keygen**, proporcionada como parte del paquete 'openssh', para crear el par de claves pública/privada.

De forma predeterminada, sin argumentos, el mandato **ssh-keygen** solicita una ubicación y frase de contraseña para los dos archivos de claves, que toma como valor predeterminado los nombres:

```
id_rsa      <-- Private key
id_rsa.pub  <-- Public key
```



**Atención:** Si está utilizando el mandato **ssh-keygen** de una versión reciente de OpenSSH, como por ejemplo la que se suministra con RHEL 8, el formato de clave utilizado no es compatible con el agente de puente de protocolo y los intentos de transferencia al servidor SFTP fallan con el mensaje:

```
BFGBR0216E: Authentication to protocol server 'sftp.host.address' failed
because of invalid private key.
```

Para crear una clave privada compatible con estas versiones más recientes de OpenSSH, especifique el formato de clave con el siguiente argumento en el mandato **ssh-keygen**:

```
ssh-keygen -m PEM
```



El contenido de la clave privada `id_rsa` tiene las primeras y últimas líneas de:

```
-----BEGIN RSA PRIVATE KEY-----
... ..
-----END RSA PRIVATE KEY-----
```

que es compatible con el agente de puente de protocolo.

2. Copie todo el contenido del archivo `id_rsa.pub` en el archivo `~/.ssh/authorized_keys` del usuario SFTP en el servidor SFTP.

Asegúrese de que los permisos de archivo de este archivo y del directorio `~/.ssh` se han establecido correctamente para que el servidor SFTP permita la autenticación de claves. Estos permisos suelen ser:

```
~/.ssh          Mode 700
~/.ssh/authorized_keys  Mode 600
```

3. Managed File Transfer requiere una huella dactilar ssh de host generada mediante el algoritmo MD5. Ejecute uno de los mandatos siguientes para obtener la huella digital ssh del host del servidor SFTP.

- Para Red Hat® Enterprise Linux versión 6.x y posteriores, y Linux Ubuntu 14.04, ejecute el mandato siguiente:

```
ssh-keygen -l -f /etc/ssh/ssh_host_rsa_key.pub
```

- A partir de Red Hat Enterprise Linux versión 7.x, Linux Ubuntu 16.04 y SuSE Linux 12.4, el mandato `ssh-keygen` genera, de forma predeterminada, la huella digital ssh utilizando el algoritmo SHA56. Para generar la huella digital ssh utilizando el algoritmo MD5, ejecute el mandato siguiente:

```
ssh-keygen -l -E MD5 -f /etc/ssh/ssh_host_rsa_key.pub
```

La salida del mandato será parecida a la del ejemplo siguiente:

```
2048 MD5:64:39:f5:49:41:10:55:d2:0b:81:42:5c:87:62:9d:27 no comment (RSA)
```

Extraiga la parte hexadecimal sólo de la salida que se utilizará como clave de host en el archivo `ProtocolBridgeCredentials.xml` (consulte el paso “4” en la página 297). Por lo tanto, en este ejemplo, se extraerá `64:39:f5:49:41:10:55:d2:0b:81:42:5c:87:62:9d:27`.

4. En el sistema de agente de puente de protocolo, edite el archivo `ProtocolBridgeCredentials.xml`. Sustituya los valores que aparecen en cursiva en el ejemplo siguiente por sus propios valores:

```
<tns:credentials xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeCredentials"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeCredentials
ProtocolBridgeCredentials.xsd">
  <tns:agent name="Agent_name">
  <tns:server name="SFTP_name">
  <tns:user name="mq_User_ID" serverUserId="SFTP_user_ID"
  hostKey="ssh_host_finger">
  <tns:privateKey associationName="name" keyPassword="pass_phrase">
  Complete contents of the id_rsa file including the entries
  -----BEGIN RSA PRIVATE KEY-----
  -----END RSA PRIVATE KEY-----
  </tns:privateKey>
  </tns:user>
  </tns:server>
  </tns:agent>
</tns:credentials>
```

donde:

- *nombre\_agente* es el nombre del agente de puente de protocolo.
- *SFTP\_host\_name* es el nombre del servidor SFTP tal como se muestra en el archivo `ProtocolBridgeProperties.xml`.
- *ID\_usuario\_mq* es el ID de usuario MQMD asociado a la solicitud de transferencia.
- *SFTP\_user\_ID* es el ID de usuario SFTP que se utiliza en el paso 2. Es el valor que se pasa al servicio SFTP como ID de usuario de inicio de sesión.
- *huella\_host\_ssh* es la huella digital que se ha obtenido en el paso 3.
- *nombre* es un nombre que puede especificar para utilizarlo con fines de rastreo y registro cronológico.
- *frase\_contraseña* es la frase de contraseña que ha proporcionado en el mandato `ssh-keygen` en el paso 1.
- *Contenido completo del archivo id\_rsa* es el contenido completo del archivo `id_rsa` generado del paso 1. Para evitar un error de conexión, asegúrese de incluir las dos entradas siguientes:

```
-----BEGIN RSA PRIVATE KEY-----
-----END RSA PRIVATE KEY-----
```

Puede añadir claves adicionales duplicando el elemento `<tns:privatekey>`.

5. Inicie el agente de puente de protocolo si aún no está iniciado. De forma alternativa, el agente de puente de protocolo sondea periódicamente el archivo `ProtocolBridgeCredentials.xml` y recoge los cambios.

## Configurar un puente de protocolo para un servidor FTPS

Configure un servidor FTPS de forma similar a como configura un servidor FTP: cree un agente de puente para el servidor, defina las propiedades del servidor y correlacione credenciales de usuario.

### Acerca de esta tarea

Para configurar un servidor FTPS, realice los pasos siguientes:

### Procedimiento

1. Cree un agente de puente de protocolo para el servidor FTPS utilizando el mandato **fteCreateBridgeAgent**. Los parámetros que son aplicables a FTP también son aplicables a FTPS pero además hay tres parámetros obligatorios específicos de FTPS:
  - a) El parámetro **-bt** . Especifique FTPS como el valor de este parámetro.
  - b) El parámetro **-bts** para el archivo de almacén de confianza. El mandato presupone que solamente es necesaria la autenticación de servidor y debe especificar la ubicación del archivo de almacén de confianza.

La forma explícita del protocolo FTPS se configura con el mandato **fteCreateBridgeAgent** de forma predeterminada, pero puede configurar la forma implícita cambiando el archivo de propiedades de puente de protocolo. El puente de protocolo siempre se conecta a los servidores FTPS en modalidad pasiva.

Si desea más información sobre el mandato **fteCreateBridgeAgent**, consulte [fteCreateBridgeAgent \(crear y configurar un agente de puente de protocolo MFT\)](#).

Si necesita instrucciones sobre cómo crear archivos de almacén de confianza, consulte el artículo de IBM Developer, [Configuración de la conectividad de la capa de sockets seguros en IBM WebSphere MQ File Transfer Edition](#) o consulte la información sobre keytool en la [documentación de keytool de Oracle](#).

- Defina las propiedades del servidor FTPS dentro de un elemento <ftpsServer> en el archivo de propiedades del puente de protocolo: ProtocolBridgeProperties.xml. Para obtener más información, consulte [“Definición de propiedades para servidores de archivos de protocolo utilizando el archivo ProtocolBridgeProperties.xml”](#) en la página 285. También puede habilitar la autenticación de cliente editando el archivo de propiedades de puente de protocolo. Si desea detalles de todas las opciones de configuración, consulte [Formato de archivo de propiedades del puente de protocolo](#).
- Correlacione credenciales de usuario de Managed File Transfer con credenciales de usuario del servidor FTPS, utilizando la función de correlación de credenciales predeterminada del agente de puente de protocolo o escribiendo su propia salida de usuario. Para obtener más información, consulte [“Correlación de credenciales para un servidor de archivos”](#) en la página 292.
- De forma predeterminada, el archivo de almacén de confianza está configurado con el formato JKS; si desea cambiar el formato, edite el archivo de propiedades de puente de protocolo.

### Ejemplo

A continuación se muestra una entrada de ejemplo para un servidor FTPS en el archivo de propiedades de puente de protocolo:

```
<tns:serverProperties xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeProperties"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeProperties
  ProtocolBridgeProperties.xsd">
  <tns:defaultServer name="ftpsserver.mycompany.com" />

  <tns:ftpsServer name="ftpsserver.mycompany.com" host="ftpsserver.mycompany.com" port="990"
  platform="windows"
  timeZone="Europe/London" locale="en_US" fileEncoding="UTF8"
  listFormat="unix" limitedWrite="false"
  trustStore="c:\mydirec\truststore.jks" />

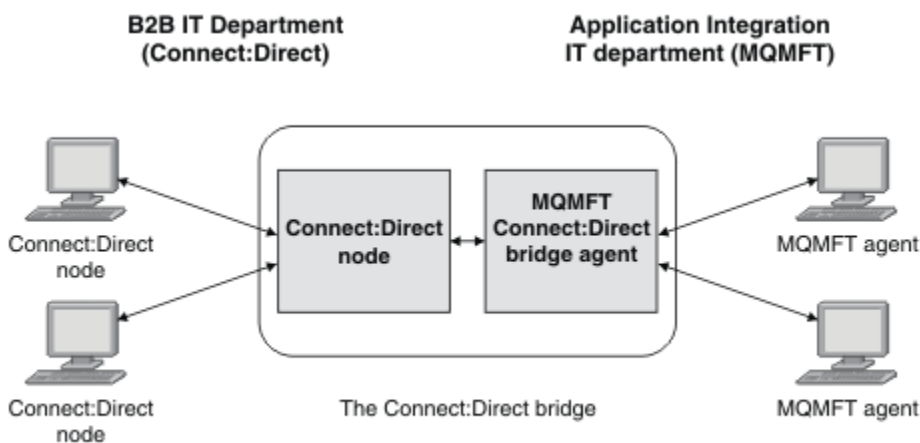
  <!-- Define servers here -->
</tns:serverProperties>
```

### Qué hacer a continuación

Si desea información sobre las partes del protocolo FTPS que están soportadas y que no están soportadas, consulte [Soporte del servidor FTPS por el puente de protocolo](#).

## El puente Connect:Direct

Puede transferir archivos a y desde una red de IBM Sterling Connect:Direct existente. Utilice el puente Connect:Direct, que es un componente de Managed File Transfer, para transferir archivos entre MFT y IBM Sterling Connect:Direct.



El diagrama muestra un puente MFT Connect:Direct entre dos departamentos, el departamento de TI de B2B y el departamento de TI de Application Integration. El departamento de TI de B2B utiliza



Connect:Direct para transferir archivos a y desde los business partners de la empresa. El departamento Application Integration IT utiliza IBM MQ como la infraestructura de mensajería y por ello ha elegido recientemente Managed File Transfer como su solución de transferencia de archivos.

Utilizando el puente MFT Connect:Direct, los dos departamentos pueden transferir archivos entre la red Connect:Direct en el departamento de TI B2B y la red MFT en el departamento de TI de integración de aplicaciones. El puente Connect:Direct es un componente de Managed File Transfer, que incluye un agente de MFT que se comunica con un nodo Connect:Direct. El agente MFT se dedica a transferencias con el agente nodo Connect:Direct y se conoce como agente de puente Connect:Direct.

El puente Connect:Direct está disponible como parte de los componentes Servicio y Agente de Managed File Transfer y se puede utilizar en las tareas siguientes:


1. Utilizar mandatos Managed File Transfer para iniciar una transferencia de un archivo o varios archivos, desde un agente de MFT a un nodo Connect:Direct.
2. Utilizar mandatos Managed File Transfer para iniciar una transferencia de un archivo o varios archivos, desde un nodo Connect:Direct a un agente de MFT.
3. Utilice los mandatos Managed File Transfer para iniciar una transferencia de archivos que inicie un proceso Connect:Direct definido por el usuario.
4. Utilizar un proceso Connect:Direct para someter una solicitud de transferencia de archivos de MFT.

Un puente Connect:Direct puede transferir archivos a o desde únicamente nodos Connect:Direct. El puente Connect:Direct puede transferir archivos a o desde su sistema de archivos local sólo como parte de una transferencia enviada por un proceso de Connect:Direct.

 Puede utilizar el puente Connect:Direct para transferir a o desde un conjunto de datos que se encuentre en un nodo Connect:Direct de un sistema z/OS. Existen algunas diferencias de comportamiento en comparación con las transferencias de conjuntos de datos en las que sólo participan agentes de Managed File Transfer. Para obtener más información, consulte  [Transferencia de conjuntos de datos a y desde nodos Connect:Direct](#).

## Plataformas soportadas

El puente Connect:Direct está formado por un agente de puente MFT Connect:Direct y un nodo Connect:Direct. El agente recibe soporte en Windows y Linux para x86-64. El nodo recibe soporte en plataformas admitidas para IBM Sterling Connect:Direct para Windows y IBM Sterling Connect:Direct para UNIX. Si desea instrucciones sobre cómo crear el agente de puente Connect:Direct y configurar un nodo Connect:Direct para que el agente se comuniquen con él, consulte [Configuración del puente Connect:Direct](#).

El puente Connect:Direct puede transferir archivos a y desde nodos Connect:Direct que se ejecuten como parte de una instalación del Servicio Connect:Direct para Windows o Connect:Direct para UNIX  o Connect:Direct para z/OS. Para obtener información de las versiones de Connect:Direct admitidas, consulte la página web [Requisitos del sistema para IBM MQ](#).

El agente y el nodo que forman el puente Connect:Direct deben estar en el mismo sistema, o tener acceso al mismo sistema de archivos, por ejemplo a través de un montaje NFS compartido. Este sistema de archivos se utiliza para almacenar temporalmente archivos durante las transferencias de archivos que implican el puente Connect:Direct, en un directorio definido por el parámetro **cdTmpDir**. El agente de puente Connect:Direct y el nodo de puente Connect:Direct deben poder acceder a este directorio utilizando el mismo nombre de vía de acceso. Por ejemplo, si el agente y el nodo están en sistemas Windows distintos, los sistemas deben utilizar la misma letra de unidad para montar el sistema de archivos compartidos. Las siguientes configuraciones permiten que el agente y el nodo utilicen el mismo nombre de vía de acceso:

- El agente y el nodo se hallan en el mismo sistema, que está ejecutando Windows o Linux para x86-64
- El agente está en Linux para x86-64 y el nodo está en UNIX
- El agente está en un sistema Windows y el nodo está en otro sistema Windows

Las siguientes configuraciones no permiten que el agente y el nodo utilicen el mismo nombre de vía de acceso:

- El agente está en Linux para x86-64 y el nodo está en Windows
- El agente está en Windows y el nodo está en UNIX

Tenga en cuenta esta restricción al planificar la instalación del puente Connect:Direct.

## Transferir un archivo a un nodo Connect:Direct

Se puede transferir un archivo desde un agente de Managed File Transfer a un nodo Connect:Direct utilizando el puente Connect:Direct. Especifique un nodo Connect:Direct como destino de la transferencia especificando el agente de puente Connect:Direct como el agente de destino y especificando el archivo de destino en el formato `connect_direct_node_name:file_path`.

### Antes de empezar

Para poder transferir un archivo, antes debe configurar el puente Connect:Direct, que es un componente de Managed File Transfer. Si desea más información, consulte [Configuración del puente Connect:Direct](#).

### Acerca de esta tarea

En este ejemplo, el agente de puente Connect:Direct se denomina CD\_BRIDGE. El agente de origen se denomina FTE\_AGENT y puede ser de cualquier versión de WMQFTE. El nodo Connect:Direct de destino se denomina CD\_NODE1. El archivo que se va a transferir se encuentra en la vía de acceso del archivo `/home/helen/file.log` en el sistema donde se encuentra FTE\_AGENT. El archivo se transfiere a la vía de acceso del archivo `/files/data.log` en el sistema donde se ejecuta CD\_NODE1.

### Procedimiento

1. Utilice el mandato `fteCreateTransfer` con el valor para el parámetro **-df** (archivo de destino) en el formato `connect_direct_node_name:file_path` y el valor del parámetro **-da** (agente de destino) especificado como el nombre del agente de puente Connect:Direct .

**Nota:** El nodo Connect:Direct especificado por `connect_direct_node_name` es el nodo al que desea transferir el archivo, no el nodo Connect:Direct que funciona como parte del puente Connect:Direct..

```
fteCreateTransfer -sa FTE_AGENT -da CD_BRIDGE
-df CD_NODE1:/files/data.log /home/helen/file.log
```

Si desea más información, consulte [fteCreateTransfer: iniciar una nueva transferencia de archivos](#).

2. El agente de origen FTE\_AGENT transfiere el archivo al agente de puente Connect:Direct CD\_BRIDGE. El archivo se almacena temporalmente en el sistema en el que se ejecuta el agente de puente Connect:Direct, en la ubicación que define la propiedad de agente `cdTmpDir`. El agente de puente Connect:Direct transfiere el archivo al nodo Connect:Direct CD\_NODE1.

### Conceptos relacionados

“El puente Connect:Direct” en la [página 299](#)

Puede transferir archivos a y desde una red de IBM Sterling Connect:Direct existente. Utilice el puente Connect:Direct, que es un componente de Managed File Transfer, para transferir archivos entre MFT y IBM Sterling Connect:Direct.

### Tareas relacionadas

“Transferir un archivo desde un nodo Connect:Direct” en la [página 302](#)

Se puede transferir un archivo desde un nodo Connect:Direct a un Managed File Transfer Agent utilizando el puente Connect:Direct. Puede especificar un nodo Connect:Direct como origen de la transferencia especificando el agente de puente Connect:Direct como el agente de origen y especificando la especificación de origen en el formato `connect_direct_node_name:file_path`.

## Referencia relacionada

El archivo `MFT.agent.properties`

## Transferir un archivo desde un nodo Connect:Direct

Se puede transferir un archivo desde un nodo Connect:Direct a un Managed File Transfer Agent utilizando el puente Connect:Direct. Puede especificar un nodo Connect:Direct como origen de la transferencia especificando el agente de puente Connect:Direct como el agente de origen y especificando la especificación de origen en el formato `connect_direct_node_name:file_path`.

## Antes de empezar

Para poder transferir un archivo, antes debe configurar el puente Connect:Direct, que es un componente de Managed File Transfer. Consulte [Configuración del puente Connect:Direct](#).

## Acerca de esta tarea

En este ejemplo, el agente de puente Connect:Direct se denomina CD\_BRIDGE. El agente de destino se denomina FTE\_AGENT y puede ser de cualquier versión de Managed File Transfer. El nodo Connect:Direct de origen se denomina CD\_NODE1. El archivo que se va a transferir se encuentra en la vía de acceso del archivo `/home/brian/in.file` en el sistema donde se encuentra CD\_NODE1. El archivo se transfiere a la vía de acceso del archivo `/files/out.file` en el sistema en el que se está ejecutando FTE\_AGENT.

## Procedimiento

Utilice el mandato **fteCreateTransfer** con el valor para la especificación de origen en el formato `connect_direct_node_name:file_path` y el valor del parámetro **-sa** especificado como el nombre del agente de puente Connect:Direct .

**Nota:** El nodo Connect:Direct especificado por `connect_direct_node_name` es el nodo al que desea transferir el archivo, no el nodo Connect:Direct que opera como parte del puente Connect:Direct. Por ejemplo:

```
fteCreateTransfer -sa CD_BRIDGE -da FTE_AGENT
                 -df /files/out.file CD_NODE1:/home/brian/in.file
```

Si desea más información, consulte [fteCreateTransfer: iniciar una nueva transferencia de archivos](#).

## Resultados

El agente de puente Connect:Direct CD\_BRIDGE solicita el archivo del nodo Connect:Direct CD\_NODE1. El nodo Connect:Direct envía el archivo al puente Connect:Direct. Mientras el archivo se transfiere desde el nodo Connect:Direct, el puente Connect:Direct almacena el archivo temporalmente en la ubicación definida por la propiedad de agente `cdTmpDir`. Cuando el archivo ha terminado de transferirse desde el nodo Connect:Direct al puente Connect:Direct, el puente Connect:Direct envía el archivo al agente de destino FTE\_AGENT y luego suprime el archivo de la ubicación temporal.

## Conceptos relacionados

“El puente Connect:Direct” en la página 299

Puede transferir archivos a y desde una red de IBM Sterling Connect:Direct existente. Utilice el puente Connect:Direct, que es un componente de Managed File Transfer, para transferir archivos entre MFT y IBM Sterling Connect:Direct.

## Referencia relacionada

El archivo `MFT.agent.properties`

## Transferencia de un conjunto de datos a un nodo Connect:Direct en z/OS

Se puede transferir un conjunto de datos desde un agente de Managed File Transfer en z/OS a un nodo Connect:Direct en z/OS utilizando un puente Connect:Direct que se halle en un sistema Windows o Linux.

## Antes de empezar

Para poder transferir un archivo, antes debe configurar el puente Connect:Direct, que es un componente de Managed File Transfer. Consulte [Configuración del puente Connect:Direct](#).

## Acerca de esta tarea

En este ejemplo, el parámetro **-df** se utiliza para especificar el destino de la transferencia. El uso del parámetro **-df** es válido cuando el agente de origen de la transferencia es de cualquier versión de Managed File Transfer. Si el agente de origen es de la versión IBM WebSphere MQ File Transfer Edition 7.0.4 o posterior, en su lugar, puede utilizar el parámetro **-ds**. El agente de origen se denomina FTE\_ZOS1 y es un agente de IBM WebSphere MQ File Transfer Edition 7.0.3. El agente de puente de Connect:Direct se denomina CD\_BRIDGE y se halla en un sistema Linux. El nodo Connect:Direct de destino se denomina CD\_ZOS2. El nodo Connect:Direct tanto del agente de origen como del agente de destino se hallan en los sistemas z/OS. El conjunto de datos que se va a transferir se encuentra en //FTEUSER.SOURCE.LIB en el sistema donde se encuentra FTE\_ZOS1. El conjunto de datos se transfiere al conjunto de datos //CDUSER.DEST.LIB en el sistema donde se encuentra CD\_ZOS2.

## Procedimiento

1. Utilice el mandato `fteCreateTransfer` con el valor para el parámetro **-df** con el formato: `connect_direct_node_name:data_set_name;attributes` y el valor del parámetro **-da** (agente de destino) especificado como el nombre del agente de puente Connect:Direct.

El nodo Connect:Direct que especifica `connect_direct_node_name` es el nodo al que desea transferir el conjunto de datos, no el nodo Connect:Direct que opera como parte del puente Connect:Direct.

El nombre de conjunto de datos especificado mediante `nombre_conjunto_datos` debe ser absoluto, no relativo. Connect:Direct no antepone el nombre del usuario como prefijo del nombre de conjunto de datos.

```
fteCreateTransfer -sa FTE_ZOS1 -sm QM_ZOS
                 -da CD_BRIDGE -dm QM_BRIDGE
                 -df CD_ZOS2: //'CDUSER.DEST.LIB;BLKSIZE(8000);LRECL(80)'
                 //'FTEUSER.SOURCE.LIB'
```

Si desea más información, consulte **[fteCreateTransfer](#)**: iniciar una nueva transferencia de archivos.


2. El agente de origen FTE\_ZOS1 transfiere los datos del conjunto de datos al agente de puente Connect:Direct CD\_BRIDGE. Los datos se almacenan temporalmente en un archivo sin formato en el sistema donde se ejecuta el agente de puente Connect:Direct, en la ubicación definida por la propiedad de agente `cdTmpDir`. El agente de puente Connect:Direct transfiere los datos al nodo Connect:Direct CD\_ZOS2. Cuando se completa la transferencia, el archivo sin formato se suprime del sistema en el que se ejecuta el agente de puente Connect:Direct.

## Conceptos relacionados

“El puente Connect:Direct” en la página 299

Puede transferir archivos a y desde una red de IBM Sterling Connect:Direct existente. Utilice el puente Connect:Direct, que es un componente de Managed File Transfer, para transferir archivos entre MFT y IBM Sterling Connect:Direct.

## Tareas relacionadas

 [Transferencia de conjuntos de datos a y desde nodos Connect:Direct](#)

## Referencia relacionada

 [Propiedades de BPXWDYN que no se deben utilizar con MFT](#)

## Transferencia de varios archivos a un nodo Connect:Direct

Se pueden transferir varios archivos desde un Managed File Transfer Agent a un nodo Connect:Direct utilizando el puente Connect:Direct. Para utilizar un nodo Connect:Direct como destino de la transferencia



de varios archivos, especifique el agente de puente Connect:Direct como agente de destino y especifique el directorio de destino en el formato `connect_direct_node_name:directory_path`.

## Antes de empezar

Para poder transferir archivos, antes debe configurar el puente Connect:Direct, que es un componente de Managed File Transfer. Consulte [Configuración del puente Connect:Direct](#).

## Acerca de esta tarea

En este ejemplo, el agente de origen se denomina FTE\_AGENT. El agente de puente Connect:Direct se denomina CD\_BRIDGE. El nodo Connect:Direct de destino se denomina CD\_NODE1. Los archivos que se van a transferir son `/home/jack/data.log`, `/logs/log1.txt` y `/results/latest` en el sistema donde se encuentra FTE\_AGENT. Los archivos se transfieren al directorio `/in/files` del sistema donde se ejecuta CD\_NODE1.

## Procedimiento

Utilice el mandato `fteCreateTransfer` con el valor del parámetro **-dd** (directorio de destino) en el formato `connect_direct_node_name:directory_path`. Especifique el valor del parámetro **-da** (agente de destino) como nombre del agente de puente Connect:Direct.

**Nota:** El nodo Connect:Direct que especifica `connect_direct_node_name` es el nodo al que desea que se transfieran los archivos, no el nodo Connect:Direct que funciona como parte del puente Connect:Direct..

```
fteCreateTransfer -sa FTE_AGENT -da CD_BRIDGE
                  -dd CD_NODE1:/in/files /home/jack/data.log
                  /logs/log1.txt /results/latest
```

Si desea más información, consulte [fteCreateTransfer: iniciar una nueva transferencia de archivos](#).

## Resultados

El agente de origen FTE\_AGENT transfiere el primer archivo al agente de puente Connect:Direct CD\_BRIDGE. El agente de puente Connect:Direct almacena temporalmente el archivo en la ubicación definida por la propiedad `cdTmpDir`. Cuando el archivo se ha transferido por completo desde el agente de origen al puente Connect:Direct, el agente de puente Connect:Direct envía el archivo al nodo Connect:Direct que define la propiedad de agente `cdNode`. Este nodo envía el archivo al nodo Connect:Direct de destino CD\_NODE1. El agente de puente Connect:Direct suprime el archivo de la ubicación temporal cuando finaliza la transferencia entre los nodos Connect:Direct. Este proceso se repite para cada archivo de origen especificado.

### Conceptos relacionados

[“El puente Connect:Direct” en la página 299](#)

Puede transferir archivos a y desde una red de IBM Sterling Connect:Direct existente. Utilice el puente Connect:Direct, que es un componente de Managed File Transfer, para transferir archivos entre MFT y IBM Sterling Connect:Direct.

### Tareas relacionadas

[“Transferir un archivo a un nodo Connect:Direct” en la página 301](#)

Se puede transferir un archivo desde un agente de Managed File Transfer a un nodo Connect:Direct utilizando el puente Connect:Direct. Especifique un nodo Connect:Direct como destino de la transferencia especificando el agente de puente Connect:Direct como el agente de destino y especificando el archivo de destino en el formato `connect_direct_node_name:file_path`.

[“Transferencia de varios archivos a Connect:Direct utilizando comodines” en la página 306](#)

Para transferir varios archivos desde un agente de Managed File Transfer a un nodo Connect:Direct, utilice el puente Connect:Direct. Puede utilizar caracteres comodín en la especificación de origen que suministra en el mandato **fteCreateTransfer**. Al igual que todas las transferencias de Managed File Transfer con comodines, sólo la última parte de la vía de acceso al archivo puede contener un carácter comodín. Por ejemplo, `/abc/def*` es una vía de acceso de archivo válida y `/abc*/def` no es válida.



[“Transferir un archivo desde un nodo Connect:Direct” en la página 302](#)

Se puede transferir un archivo desde un nodo Connect:Direct a un Managed File Transfer Agent utilizando el puente Connect:Direct. Puede especificar un nodo Connect:Direct como origen de la transferencia especificando el agente de puente Connect:Direct como el agente de origen y especificando la especificación de origen en el formato `connect_direct_node_name:file_path`.

[“Transferir varios archivos desde un nodo Connect:Direct” en la página 305](#)

Se pueden transferir múltiples archivos desde un nodo Connect:Direct a un Managed File Transfer Agent utilizando el puente Connect:Direct. Puede especificar un nodo Connect:Direct como origen de la transferencia de varios archivos especificando el agente de puente Connect:Direct como el agente de origen y especificando una o más especificaciones de origen en el formato `connect_direct_node_name:file_path`.

### Referencia relacionada

El archivo MFT `agent.properties`

## Transferir varios archivos desde un nodo Connect:Direct

Se pueden transferir múltiples archivos desde un nodo Connect:Direct a un Managed File Transfer Agent utilizando el puente Connect:Direct. Puede especificar un nodo Connect:Direct como origen de la transferencia de varios archivos especificando el agente de puente Connect:Direct como el agente de origen y especificando una o más especificaciones de origen en el formato `connect_direct_node_name:file_path`.

### Antes de empezar

Para poder transferir un archivo, antes debe configurar el puente Connect:Direct, que es un componente de Managed File Transfer. Consulte [Configuración del puente Connect:Direct](#).

### Acerca de esta tarea

En este ejemplo, el agente de puente Connect:Direct se denomina CD\_BRIDGE. El agente de destino se denomina FTE\_Z y se ejecuta en un sistema z/OS. El nodo Connect:Direct de origen se denomina CD\_NODE1. Los archivos que se van a transferir se encuentran en las vías de acceso de archivo `/in/file1`, `/in/file2` y `/in/file3` en el sistema donde se encuentra CD\_NODE1. Los archivos se transfieren al conjunto de datos particionados `//OBJECT.LIB` en el sistema en el que se ejecuta FTE\_Z.

### Procedimiento

Utilice el mandato `fteCreateTransfer` con los valores para las especificaciones de origen en el formato `connect_direct_node_name:file_path` y el valor del parámetro `-sa` especificado como el nombre del agente de puente Connect:Direct .

**Nota:** El nodo Connect:Direct especificado por `connect_direct_node_name` es el nodo al que desea que se transfieran los archivos, no el nodo Connect:Direct que funciona como parte del puente Connect:Direct.

```
fteCreateTransfer -sa CD_BRIDGE -da FTE_Z
                  -dp '//OBJECT.LIB' CD_NODE1:/in/file1
                  CD_NODE1:/in/file2 CD_NODE1:/in/file3
```

Si desea más información, consulte [fteCreateTransfer](#): iniciar una nueva transferencia de archivos.

### Resultados

El agente de puente Connect:Direct CD\_BRIDGE solicita el primer archivo del nodo Connect:Direct CD\_NODE1. El nodo Connect:Direct envía el archivo al puente Connect:Direct. Mientras el archivo se transfiere desde el nodo Connect:Direct, el puente Connect:Direct almacena el archivo temporalmente en la ubicación definida por la propiedad de agente `cdTmpDir`. Cuando el archivo ha terminado de transferirse desde el nodo Connect:Direct al puente Connect:Direct, el puente Connect:Direct envía el archivo al agente de destino FTE\_Z y, a continuación, lo suprime de la ubicación temporal. Este proceso se repite para cada archivo de origen especificado.

## Conceptos relacionados

“El puente Connect:Direct” en la página 299

Puede transferir archivos a y desde una red de IBM Sterling Connect:Direct existente. Utilice el puente Connect:Direct, que es un componente de Managed File Transfer, para transferir archivos entre MFT y IBM Sterling Connect:Direct.

## Referencia relacionada

El archivo `MFT_agent.properties`

## Transferencia de varios archivos a Connect:Direct utilizando comodines

Para transferir varios archivos desde un agente de Managed File Transfer a un nodo Connect:Direct, utilice el puente Connect:Direct. Puede utilizar caracteres comodín en la especificación de origen que suministra en el mandato **fteCreateTransfer**. Al igual que todas las transferencias de Managed File Transfer con comodines, sólo la última parte de la vía de acceso al archivo puede contener un carácter comodín. Por ejemplo, `/abc/def*` es una vía de acceso de archivo válida y `/abc*/def` no es válida.

## Antes de empezar

Para poder transferir un archivo, antes debe configurar el puente Connect:Direct, que es un componente de Managed File Transfer. Si desea más información, consulte [Configuración del puente Connect:Direct](#).

## Acerca de esta tarea

En este ejemplo, el agente de origen se denomina FTE\_AGENT y el agente de puente Connect:Direct se denomina CD\_BRIDGE. El nodo Connect:Direct de destino se denomina CD\_NODE1. Los archivos que se van a transferir se encuentran en el directorio `/reports` del sistema donde se encuentra FTE\_AGENT. Sólo se transfieren los archivos con nombres que empiezan por `report`, seguidos de dos caracteres y el sufijo `.log`. Por ejemplo, se transfiere el archivo `/reports/report01.log`, pero el archivo `/reports/report1.log` no se transfiere. Los archivos se transfieren al directorio `/home/fred` del sistema donde se ejecuta CD\_NODE1.

## Procedimiento

1. Utilice el mandato `fteCreateTransfer` con el valor del parámetro **-dd** (directorio de destino) en el formato `connect_direct_node_name:directory_path`. Para el parámetro **-da** (agente de destino), especifique el agente de puente Connect:Direct.

**Nota:** El nodo Connect:Direct que especifica `connect_direct_node_name` es el nodo al que desea que se transfieran los archivos, no el nodo Connect:Direct que funciona como parte del puente Connect:Direct..

```
fteCreateTransfer -sa FTE_AGENT -da CD_BRIDGE
                 -dd CD_NODE1:/home/fred "/reports/report??.log"
```

Si desea más información, consulte [fteCreateTransfer: iniciar una nueva transferencia de archivos](#).

2. El agente de origen FTE\_AGENT transfiere el primer archivo que coincide con el patrón de `/reports/report??.log` con el agente de puente Connect:Direct CD\_BRIDGE. El agente de puente Connect:Direct almacena temporalmente el archivo en la ubicación definida por la propiedad `cdTmpDir`. Cuando el archivo se ha transferido por completo desde el agente de origen al puente Connect:Direct, el agente de puente Connect:Direct envía el archivo al nodo Connect:Direct que define la propiedad de agente `cdNode`. Este nodo envía el archivo al nodo Connect:Direct de destino CD\_NODE1. El agente de puente Connect:Direct suprime el archivo de la ubicación temporal cuando finaliza la transferencia entre los nodos Connect:Direct. Este proceso se repite para cada archivo fuente que coincida con el patrón de comodín `/reports/report??.log`.

**Nota:** La lista de archivos que coinciden con el patrón de `/reports/report??.log` varía en función del sistema operativo del sistema en el que se encuentra el agente de origen FTE\_AGENT.

- Si el agente de origen se encuentra en un sistema con un sistema operativo Windows, la coincidencia del patrón no distingue entre mayúsculas y minúsculas. El patrón coincide con todos los archivos del directorio `/reports` con un nombre de archivo del formato `report` seguido de dos caracteres y un sufijo de `.log`, independientemente de las mayúsculas y minúsculas en el que se encuentren las letras. Por ejemplo, `Report99.Log` es una coincidencia.
- Si el agente de origen se halla en un sistema con un sistema operativo Linux o UNIX, la coincidencia del patrón distingue entre mayúsculas y minúsculas. El patrón sólo coincide con los archivos del directorio `/reports` con un nombre de archivo del formato `report` seguido de dos caracteres y un sufijo de `.log`. Por ejemplo, `reportAB.log` es una coincidencia, pero `reportAB.LOG` y `Report99.Log` no son coincidencias.

## Recuperación y reinicio de transferencias a y desde nodos Connect:Direct

Managed File Transfer podría no ser capaz de conectarse al nodo IBM Sterling Connect:Direct durante una transferencia; por ejemplo, si el nodo deja de estar disponible. Managed File Transfer intenta recuperar la transferencia, o la transferencia falla y se genera un mensaje de error.

### Si el nodo Connect:Direct no está disponible

Si el nodo Connect:Direct deja de estar disponible; por ejemplo, debido a una interrupción de la alimentación o de la red, Managed File Transfer recupera una transferencia de archivos de las siguientes maneras:

- Si Managed File Transfer no se ha conectado correctamente anteriormente al nodo Connect:Direct como parte de esta solicitud de transferencia, la transferencia se vuelve a intentar durante un periodo de tiempo determinado por los valores de **`cdMaxConnectionRetries`** y **`recoverableTransferRetryInterval`** **properties**. Estas propiedades se especifican en el archivo `agent.properties` para el agente de puente Connect:Direct. La transferencia falla y se genera un mensaje de error, después de que el número de intentos fallidos alcance el valor de **`cdMaxConnectionRetries`** **property**. De forma predeterminada, se intenta la transferencia indefinidamente, con una espera de 60 segundos entre cada intento.
- Si Managed File Transfer ya se ha conectado correctamente al nodo Connect:Direct como parte de esta solicitud de transferencia, se intenta de nuevo la transferencia durante un tiempo determinado por los valores de las propiedades **`cdMaxPartialWorkConnectionRetries`** y **`recoverableTransferRetryInterval`**. La transferencia falla, y se genera un mensaje de error, después de que el número de intentos anómalos alcance el valor de la propiedad **`cdMaxPartialWorkConnectionRetries`**. De forma predeterminada, se intenta la transferencia indefinidamente, con una espera de 60 segundos entre cada intento.
- Para determinados tipos de anomalías de nodo Connect:Direct, por ejemplo, si el nodo se detiene de forma forzosa, los procesos Connect:Direct pasan a tener el estado `Held Due to Error (HE)` cuando se recupera el nodo. Una vez que se recupera el nodo, Managed File Transfer reanuda automáticamente cualquier proceso Connect:Direct que esté relacionado con la transferencia de archivos y tenga un estado de HE.
- Si la transferencia falla, los archivos temporales relacionados con la transferencia se suprimen del sistema que aloja el puente Connect:Direct. La ubicación de estos archivos temporales está definida por la propiedad **`cdTmpDir`**.
- Si la transferencia es de Managed File Transfer a Connect:Direct y se especifica una disposición de origen de supresión, los archivos de origen no se suprimen si la transferencia falla.

### Si las credenciales de usuario del nodo Connect:Direct no son válidas

Si Managed File Transfer no puede conectarse al nodo Connect:Direct porque las credenciales del usuario son rechazadas por el nodo, la transferencia falla y se genera un mensaje de error. En esta situación, compruebe que ha proporcionado las credenciales de usuario correctas para el nodo Connect:Direct. Si desea más información, consulte [Correlación de credenciales para Connect:Direct](#).

## Si el agente de puente Connect:Direct no está disponible

Si el agente de puente Connect:Direct deja de estar disponible, las transferencias de archivos en curso se recuperan de la misma manera que las transferencias de Managed File Transfer estándar. Para obtener más información, consulte [“Recuperación y reinicio de MFT” en la página 314](#).

### Conceptos relacionados

[“El puente Connect:Direct” en la página 299](#)

Puede transferir archivos a y desde una red de IBM Sterling Connect:Direct existente. Utilice el puente Connect:Direct, que es un componente de Managed File Transfer, para transferir archivos entre MFT y IBM Sterling Connect:Direct.

[“Recuperación y reinicio de MFT” en la página 314](#)

Si el agente o el gestor de colas están no disponibles por algún motivo, por ejemplo, debido a un fallo en el suministro o en la red, Managed File Transfer se recupera tal como se indica a continuación en los siguientes escenarios:

### Tareas relacionadas

[Configurar el puente Connect:Direct](#)

### Referencia relacionada

El archivo MFT `agent.properties`

## Cómo someter un proceso Connect:Direct definido por el usuario desde una solicitud de transferencia de archivos

Puede someter una solicitud de transferencia para una transferencia que pasa a través del agente de puente Connect:Direct que llame a un proceso Connect:Direct definido por el usuario como parte de la transferencia de archivos.

De forma predeterminada, cuando se somete una solicitud de transferencia de archivos para una transferencia que pasa a través del agente de puente Connect:Direct, el agente de puente Connect:Direct genera el proceso Connect:Direct que se utiliza para transferir el archivo a o desde el nodo Connect:Direct remoto.

Sin embargo, puede configurar el agente de puente Connect:Direct para que en su lugar llame a un proceso definido por el usuario Connect:Direct utilizando el archivo `ConnectDirectProcessDefinition.xml`.

## El archivo `ConnectDirectProcessDefinition.xml`

El mandato `fteCreateCDAgent` crea el archivo `ConnectDirectProcessDefinitions.xml` en el directorio de configuración del agente `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent_name`. Antes de poder llamar a procesos Connect:Direct definidos por el usuario desde el agente de puente Connect:Direct, debe configurar definiciones de proceso editando este archivo.

El archivo define uno o más conjuntos de procesos que incluyen la ubicación de uno o más procesos Connect:Direct a los que se llama como parte de una transferencia. Cada conjunto de procesos incluye una serie de condiciones. Si la transferencia cumple todas las condiciones del conjunto de procesos, el conjunto de procesos se utiliza para especificar a qué procesos Connect:Direct llama la transferencia. Para obtener más información, consulte [“Especificación del proceso Connect:Direct que se debe iniciar utilizando el archivo `ConnectDirectProcessDefinition.xml`” en la página 309](#).

## Variables simbólicas intrínsecas

Puede utilizar las variables simbólicas intrínsecas definidas por Managed File Transfer para sustituir valores en procesos Connect:Direct definidos por el usuario. Para seguir el convenio de denominación de Connect:Direct, todas las variables simbólicas intrínsecas utilizadas por Managed File Transfer tienen el formato `%FTE` seguido de cinco caracteres alfanuméricos en mayúsculas.

Al crear un proceso para transferir archivos desde un nodo Connect:Direct al sistema del puente Connect:Direct, debe utilizar la variable intrínseca %FTETFILE como el valor de TO FILE en el proceso Connect:Direct. Al crear un proceso para transferir archivos a un nodo Connect:Direct desde el sistema del puente Connect:Direct, debe utilizar la variable intrínseca %FTEFFILE como el valor de FROM FILE en el proceso Connect:Direct. Estas variables contienen las vías de acceso de archivos temporales que el agente de puente Connect:Direct utiliza para las transferencias dentro y fuera de la red de Managed File Transfer.

Para obtener más información sobre las variables simbólicas intrínsecas, consulte la documentación del producto Connect:Direct.

## Procesos Connect:Direct de ejemplo

Managed File Transfer proporciona procesos de Connect:Direct de ejemplo. Estos ejemplos se encuentran en el directorio siguiente: *MQ\_INSTALLATION\_PATH/mqft/samples/ConnectDirectProcessTemplates*.

### **Especificación del proceso Connect:Direct que se debe iniciar utilizando el archivo *ConnectDirectProcessDefinition.xml***

Especifique qué proceso Connect:Direct hay que iniciar como parte de una transferencia de Managed File Transfer. Managed File Transfer proporciona un archivo XML que puede editar para especificar definiciones de proceso.

## Acerca de esta tarea

El mandato **fteCreateCDAgent** crea el archivo *ConnectDirectProcessDefinitions.xml* en el directorio de configuración del agente *MQ\_DATA\_PATH/mqft/config/coordination\_queue\_manager/agents/cd\_bridge\_agent\_name*. Antes de poder llamar a procesos Connect:Direct definidos por el usuario desde el agente de puente Connect:Direct, debe configurar definiciones de proceso editando este archivo.

Para cada proceso que desee especificar para llamarlo como parte de una transferencia a través del puente Connect:Direct, realice los pasos siguientes:

## Procedimiento

1. Defina el proceso Connect:Direct al que desea que el agente de puente Connect:Direct llame como parte de la transferencia y guarde la plantilla de proceso en un archivo.
2. Abra el archivo *MQ\_DATA\_PATH/mqft/config/coordination\_queue\_manager/agents/cd\_bridge\_agent\_name/ConnectDirectProcessDefinitions.xml* en un editor de texto.
3. Cree un elemento `<processSet>`.
4. Dentro del elemento `<processSet>`, cree un elemento `<condition>`.
5. Dentro del elemento `<condition>`, cree uno o más elementos que definan una condición de que la solicitud de transferencia debe coincidir para llamar al proceso de Connect:Direct que ha definido en el Paso 1. Estos elementos pueden ser elementos `<match>` o elementos `<defined>`.
  - Utilice un elemento `<match>` para especificar que el valor de una variable debe coincidir con un patrón. Cree el elemento `<match>` con los atributos siguientes:
    - `variable` - el nombre de la variable cuyo valor se compara. La variable es un símbolo intrínseco. Si desea más información, consulte [Variables de sustitución para su uso con procesos Connect:Direct definidos por el usuario](#).
    - `value` - el patrón con el que comparar el valor de la variable especificada.
    - Opcional: `pattern` - el tipo de patrón utilizado por el valor del atributo `value`. Este tipo de patrón puede ser `wildcard` o `regex`. Este atributo es opcional y el valor predeterminado es `wildcard`.

- Utilice un elemento <defined> para especificar que una variable debe tener un valor definido. Cree el elemento <defined> con el atributo siguiente:
  - `variable` - el nombre de la variable que debe tener un valor definido. La variable es un símbolo intrínseco. Si desea más información, consulte [Variables de sustitución para su uso con procesos Connect:Direct definidos por el usuario](#).

Las condiciones especificadas en el elemento <condition> se combinan con un AND lógico. Se deben cumplir todas las condiciones para que el agente de puente Connect:Direct llame al proceso especificado por este elemento <processSet>. Si no especifica un elemento <condition>, el conjunto de procesos coincide con todas las transferencias.

6. Dentro del elemento <processSet>, cree un elemento <process>.

7. Dentro del elemento <process>, cree un elemento <transfer>.

El elemento transfer especifica el proceso Connect:Direct al que el agente de puente Connect:Direct llama como parte de la transferencia. Cree el elemento <transfer> con el atributo siguiente:

- `process` - la ubicación del proceso Connect:Direct que ha definido en el paso 1. La ubicación de este archivo se especifica con una vía de acceso absoluta o relativa al directorio `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent_name`.

## Resultados

Al buscar una coincidencia de condición, el agente de puente Connect:Direct busca desde el principio del archivo al final del archivo. La primera coincidencia que se encuentra es la que se utiliza.

### Tareas relacionadas

[Configurar el puente Connect:Direct](#)

### Referencia relacionada

[Formato del archivo de definiciones de proceso Connect:Direct](#)

[fteCreateCDAgent: crear un agente de puente Connect:Direct](#)

## **Utilizar variables simbólicas intrínsecas en procesos Connect:Direct invocados por Managed File Transfer**

Puede llamar a un proceso Connect:Direct definido por el usuario desde una transferencia de Managed File Transfer y pasar información de la transferencia al proceso Connect:Direct, utilizando variables simbólicas intrínsecas en la definición de proceso.

### Acerca de esta tarea

Este ejemplo utiliza variables simbólicas intrínsecas para pasar información de una transferencia de Managed File Transfer a un proceso Connect:Direct definido por el usuario. Si desea más información sobre variables simbólicas intrínsecas utilizadas por Managed File Transfer, consulte [Variables de sustitución para ser utilizadas con procesos Connect:Direct definidos por el usuario](#).

En este ejemplo, el archivo se transfiere desde un Managed File Transfer Agent a un nodo de puente Connect:Direct. La primera parte de la transferencia la realiza Managed File Transfer. La segunda parte de la transferencia la realiza un proceso Connect:Direct definido por el usuario.

## Procedimiento

1. Cree un proceso Connect:Direct que utilice variables simbólicas intrínsecas.

```
%FTEPNAME PROCESS
  SNODE=%FTESNODE
  PNODEID=(%FTEPUSER,%FTEPPASS)
  SNODEID=(%FTESUSER,%FTESPASS)

COPY001 COPY
  FROM (
    FILE=%FTEFFILE
    DISP=%FTEFDISP
  )
  TO (
    FILE=%FTETFILE
```

```
)  
PEND  
DISP=%FTETDISP
```

2. Guarde este proceso en un archivo de texto en la siguiente ubicación: `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent/Example.cdp`
3. Edite el archivo `ConnectDirectProcessDefinition.xml` para incluir una regla que llame al proceso `Connect:Direct` que ha creado en el paso 1.

```
<?xml version="1.0" encoding="UTF-8"?>  
<tns:cdprocess xmlns:tns="http://wmqfte.ibm.com/ConnectDirectProcessDefinitions"  
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"  
  xsi:schemaLocation="http://wmqfte.ibm.com/  
    ConnectDirectProcessDefinitions ConnectDirectProcessDefinitions.xsd">  
  
  <tns:processSet>  
    <tns:condition>  
      <tns:match variable="%FTESNODE" value="TOBERMORY" pattern="wildcard" />  
    </tns:condition>  
    <tns:process>  
      <tns:transfer process="Example.cdp" />  
    </tns:process>  
  </tns:processSet>  
</tns:cdprocess>
```

En este ejemplo, si se envía una solicitud de transferencia al agente de puente `Connect:Direct` que tiene `TOBERMORY` como su nodo `Connect:Direct` de origen o de destino, se llama al proceso `Example.cdpConnect:Direct`.

4. Envíe una solicitud de transferencia de archivos que satisfaga las condiciones que ha definido en el archivo `ConnectDirectProcessDefinition.xml` en el Paso 3.

Por ejemplo,

```
fteCreateTransfer -sa ORINOCO -da CD_BRIDGE  
                 -sm QM_WIMBLEDON -dm QM_COMMON  
                 -de overwrite -df TOBERMORY:/home/bulgaria/destination.txt  
                 -sd leave c:\bungo\source.txt
```

En este ejemplo, el nodo `Connect:Direct` de destino es `TOBERMORY`. Este nodo es el nodo secundario de la transferencia y el valor de `%FTESNODE` se establece en `TOBERMORY`. Este mandato coincide con la condición establecida en el archivo `ConnectDirectProcessDefinition.xml`.

5. `Managed File Transfer` transfiere el archivo de origen a una ubicación temporal en el mismo sistema que el agente de puente `Connect:Direct`.
6. El agente de puente `Connect:Direct` establece los valores de las variables simbólicas intrínsecas a partir de la información de la solicitud de transferencia y la información de configuración.

Las variables simbólicas intrínsecas se establecen en los siguientes valores:

- `%FTEPNAME=nombre_proceso` - Este valor es un nombre de proceso de 8 caracteres generado por el agente de puente `Connect:Direct`.
- `%FTESNODE=TOBERMORY` - Este valor se establece a partir del parámetro **-df** del mandato **fteCreateTransfer**.
- `%FTEPUSER,=primary_node_user` - Esta información se toma del archivo `ConnectDirectCredentials.xml`.
- `%FTEPPASS=primary_node_user_password` - Esta información se toma del archivo `ConnectDirectCredentials.xml`.
- `%FTESUSER,=secondary_node_user` - Esta información se toma del archivo `ConnectDirectCredentials.xml`.
- `%FTESPASS=secondary_node_user_password` - Esta información se toma del archivo `ConnectDirectCredentials.xml`.



- %FTEFFILE =*ubicación\_temporal* - Este valor es la ubicación temporal del archivo en el mismo sistema que el agente de puente Connect:Direct.
- %FTEFDISP=leave - Este valor se establece a partir del parámetro **-sd** del mandato **fteCreateTransfer**.
- %FTETFILE=/home/bulgaria/destination.txt - Este valor se establece a partir del parámetro **-df** del mandato **fteCreateTransfer**.
- %FTETDISP=overwrite - Este valor se establece a partir del parámetro **-de** del mandato **fteCreateTransfer**.

7. El proceso Connect:Direct se inicia en el nodo de puente Connect:Direct. Connect:Direct transfiere el archivo de la ubicación temporal en el sistema del puente Connect:Direct al destino /home/bulgaria/destination.txt en el sistema donde se ejecuta el nodo Connect:Direct TOBERMORY.

### Conceptos relacionados

“Cómo someter un proceso Connect:Direct definido por el usuario desde una solicitud de transferencia de archivos” en la página 308

Puede someter una solicitud de transferencia para una transferencia que pasa a través del agente de puente Connect:Direct que llame a un proceso Connect:Direct definido por el usuario como parte de la transferencia de archivos.

### Referencia relacionada

Variables de sustitución para usar con procesos Connect:Direct definidos por el usuario

## Utilización de procesos Connect:Direct para someter solicitudes de transferencia de Managed File Transfer

Puede someter una solicitud de transferencia al agente de puente Connect:Direct desde un proceso Connect:Direct. Managed File Transfer proporciona mandatos a los que se puede llamar desde una sentencia **RUN TASK** en un proceso Connect:Direct .

Managed File Transfer proporciona los siguientes mandatos para utilizarlos con procesos Connect:Direct:

### ftetag

Especifique este mandato en un paso que preceda al mandato **ftebxf** o **ftecxfer** para crear la información de auditoría necesaria para la transferencia. Este mandato acepta la especificación de origen de la transferencia como parámetro. Si desea más información sobre el formato de la especificación de origen, consulte **fteCreateTransfer**: iniciar una nueva transferencia de archivos.

### ftebxf

Especifique este mandato para crear una solicitud de transferencia de archivos cuando el gestor de colas al que se somete la solicitud de transferencia esté ubicado en el mismo sistema que el nodo Connect:Direct que somete el mandato. Este mandato acepta los mismos parámetros que el mandato **fteCreateTransfer**. Si desea más información sobre estos parámetros, consulte **fteCreateTransfer**: iniciar una nueva transferencia de archivos. Este mandato tiene también un parámetro adicional:

#### **-qmgrname**

Obligatorio. El nombre del gestor de colas al que someter el mandato.

### ftecxfer

Especifique este mandato para crear una solicitud de transferencia de archivos cuando el gestor de colas al que se somete la solicitud de transferencia no esté ubicado en el mismo sistema que el nodo Connect:Direct que somete el mandato. Este mandato acepta los mismos parámetros que el mandato **fteCreateTransfer**. Si desea información sobre los parámetros, consulte **fteCreateTransfer**: iniciar una nueva transferencia de archivos. Este mandato tiene también tres parámetros adicionales:

#### **-qmgrname**

Obligatorio. El nombre del gestor de colas al que someter el mandato.

#### **-conname**

Obligatorio. El host y el puerto del gestor de colas al que someter el mandato, especificados en formato CONNAME de IBM MQ. Por ejemplo, host.example.com(1337).



### **-channelname**

Opcional. El nombre del canal que se va a utilizar para conectarse al gestor de colas al que someter el mandato. Si no se especifica este parámetro, se utiliza un valor predeterminado de SYSTEM.DEF.SVRCONN.

### **Tareas relacionadas**

“Crear y someter un proceso Connect:Direct que llame a Managed File Transfer utilizando Connect:Direct Requester” en la página 313

Connect:Direct Requester es una interfaz gráfica de usuario que puede utilizar para crear y someter un proceso Connect:Direct que llame a Managed File Transfer.

### **Referencia relacionada**

Ejemplo: [un archivo de proceso Connect:Direct que llama a mandatos MFT](#)

## **Crear y someter un proceso Connect:Direct que llame a Managed File Transfer utilizando Connect:Direct Requester**

Connect:Direct Requester es una interfaz gráfica de usuario que puede utilizar para crear y someter un proceso Connect:Direct que llame a Managed File Transfer.

### **Acerca de esta tarea**

Esta tarea describe cómo crear un proceso Connect:Direct que llame al mandato Managed File Transfer **ftecxfer** o el mandato **ftebxfer**. Utilice el mandato **ftecxfer** cuando el gestor de colas al que se envía la solicitud de transferencia se encuentre en un sistema distinto de el del nodo Connect:Direct que somete el mandato. Utilice el mandato **ftebxfer** cuando el gestor de colas al que se envía la solicitud de transferencia se encuentre en el mismo sistema que el del nodo Connect:Direct que somete el mandato. El mandato **ftecxfer** establece una conexión de cliente al gestor de colas de agente del agente de origen de la transferencia. Antes de llamar al mandato **ftecxfer**, debe llamar al mandato **ftetag** y pasarle la información de especificación de origen. Esto permite registrar y auditar el proceso de la misma manera que las transferencias iniciadas desde Managed File Transfer.

### **Procedimiento**

1. Inicie Connect:Direct Requester.
2. En la pestaña **Nodos** del panel, seleccione el nodo Connect:Direct que se utiliza como el nodo primario del proceso.
3. Seleccione **Archivo > Nuevo > Proceso**. Se abre la ventana **Propiedades de proceso**.
4. En el campo **Nombre:**, escriba el nombre del proceso.
5. Seleccione el nodo secundario en la lista **Nombre de > Snode:**.
6. Seleccione el sistema operativo del nodo secundario en la lista **Sistema operativo de > Snode:**.
7. Opcional: Rellene cualquier información adicional de esta ventana que necesite.
8. Pulse **Aceptar**. Se cierra la ventana **Propiedades de proceso**.
9. Cree una sentencia que ejecute el mandato Managed File Transfer **ftetag**.
  - a) En la ventana **Proceso**, pulse el botón derecho del ratón en la sentencia **End**.
  - b) Seleccione **Insertar > Run Task**. Se abre la ventana **Sentencia Run Task**.
  - c) En el campo **Etiqueta:**, escriba Tag.
  - d) En el campo **Parámetros o mandatos opcionales**, escriba `pgm(MQ_INSTALLATION_PATH/bin/ftetag) args(source_specification)`. Si desea más información sobre el formato de *especificación\_origen*, consulte [fteCreateTransfer: iniciar una nueva transferencia de archivos](#).
  - e) Pulse **Aceptar**. Se cierra la ventana **Sentencia Run Task**.
10. Cree una sentencia que ejecute los mandatos Managed File Transfer **ftecxfer** o **ftebxfer**.
  - a) En la ventana **Proceso**, pulse el botón derecho del ratón en la sentencia **End**.
  - b) Seleccione **Insertar > Run Task**. Se abre la ventana **Sentencia Run Task**.

- c) En el campo **Etiqueta:**, escriba `Transfer`.
  - d) En el campo **Parámetros o mandatos opcionales**, escriba `pgm(MQ_INSTALLATION_PATH/bin/ftecxfex) args(parameters)` o `pgm(MQ_INSTALLATION_PATH/bin/ftebxfex) args(parameters)` en función del mandato que elija. Los parámetros utilizados por los mandatos **ftecxfex** y **ftebxfex** son los mismos que los parámetros utilizados por el mandato **fteCreateTransfer**, además de algunos parámetros adicionales específicos de **ftecxfex** y **ftebxfex**. Para obtener más información, consulte **fteCreateTransfer: iniciar una nueva transferencia de archivos y "Utilización de procesos Connect:Direct para someter solicitudes de transferencia de Managed File Transfer"** en la página 312.
  - e) Pulse **Aceptar**. Se cierra la ventana **Sentencia Run Task**.
11. Opcional: Cree todas las sentencias adicionales que necesite.
  12. Someta el proceso.
    - a) Pulse el botón derecho del ratón en la ventana **Proceso**.
    - b) Seleccione **Someter**. Se abre la ventana **Adjunto Connect:Direct**.
    - c) Entre el nombre de usuario y la contraseña que va a utilizar para ejecutar el proceso.
    - d) Pulse **Aceptar**.

### Conceptos relacionados

["Utilización de procesos Connect:Direct para someter solicitudes de transferencia de Managed File Transfer" en la página 312](#)

Puede someter una solicitud de transferencia al agente de puente Connect:Direct desde un proceso Connect:Direct. Managed File Transfer proporciona mandatos a los que se puede llamar desde una sentencia **RUN TASK** en un proceso Connect:Direct .

## Cómo trabajar con MFT desde IBM Integration Bus

Puede trabajar con Managed File Transfer desde IBM Integration Bus utilizando los nodos FTEOutput y FTEInput.

- Utilice el nodo FTEInput para transferir un archivo a través de la red utilizando Managed File Transfer y luego procesar dicho archivo como parte de un flujo de bus de integración.
- Utilice el nodo FTEOutput para transferir un archivo resultado de un flujo de bus de integración a otra ubicación en la red.

Los agentes que transfieren archivos a o desde el agente de intermediario pueden estar a cualquier nivel de Managed File Transfer.

Si desea obtener más información, consulte la documentación del producto [IBM Integration Bus](#).

## Recuperación y reinicio de MFT

Si el agente o el gestor de colas están no disponibles por algún motivo, por ejemplo, debido a un fallo en el suministro o en la red, Managed File Transfer se recupera tal como se indica a continuación en los siguientes escenarios:

- Normalmente, si se produce un problema mientras se está transfiriendo un archivo, Managed File Transfer recupera y reinicia esa transferencia de archivos una vez que se ha solucionado el problema.
- Si un archivo que estaba en proceso de ser transferido se suprime o se modifica mientras el agente o el gestor de colas no están disponibles, la transferencia falla y recibirá un mensaje en el registro de transferencias que proporciona detalles sobre la anomalía.
- Si un proceso de agente falla durante una transferencia de archivos, la transferencia continuará cuando reinicie el agente.
- Si un agente pierde la conexión con el gestor de colas de agente, el agente espera mientras intenta volver a conectarse al gestor de colas. Cuando el agente se reconecta satisfactoriamente al gestor de colas, la transferencia actual continúa.

- Si por algún motivo se detiene el agente, los supervisores de recursos asociados a un agente dejan de sondear. Cuando el agente se recupera, los supervisores también se reinician y se reanuda el sondeo de recursos.
- Para una transferencia de archivos que tenga una disposición de origen de delete, si se produce una recuperación tras enviar todos los datos de un agente de origen a un agente de destino, el archivo de origen se desbloquea antes de la supresión. Este desbloqueo significa que el archivo de origen podría modificarse posiblemente antes de que se suprima el archivo. Por tanto, si se considera que no es seguro suprimir el archivo de origen, se muestra el siguiente aviso:

```
BFGTR0075W: The source file has not been deleted because it is possible that the source file was modified after the source file was transferred.
```

En este caso, verifique que el contenido del archivo de origen no se haya modificado y, a continuación, suprimalo manualmente.

Puede comprobar el estado de las transferencias en IBM MQ Explorer. Si alguna transferencia aparece como Stalled, es posible que tenga que realizar una acción correctiva porque el estado detenido indica un problema con el agente o entre los dos agentes implicados en la transferencia.

### Tareas relacionadas

[“Establecimiento de un tiempo de espera para la recuperación de transferencias estancadas” en la página 315](#)

Puede establecer un tiempo de espera de recuperación de transferencia para las transferencias de archivos estancadas que se aplique a todas las transferencias para un agente de origen. También puede establecer un tiempo de espera de recuperación de transferencia para una transferencia individual. Si establece una cantidad de tiempo específica, en segundos, durante la cual un agente de origen sigue intentando recuperar una transferencia de archivos estancada y la transferencia no se realiza correctamente cuando el agente alcanza el tiempo de espera, la transferencia falla.

## V 9.1.0 Establecimiento de un tiempo de espera para la recuperación de transferencias estancadas

Puede establecer un tiempo de espera de recuperación de transferencia para las transferencias de archivos estancadas que se aplique a todas las transferencias para un agente de origen. También puede establecer un tiempo de espera de recuperación de transferencia para una transferencia individual. Si establece una cantidad de tiempo específica, en segundos, durante la cual un agente de origen sigue intentando recuperar una transferencia de archivos estancada y la transferencia no se realiza correctamente cuando el agente alcanza el tiempo de espera, la transferencia falla.

### Acerca de esta tarea

Desde IBM MQ 9.1, puede establecer un tiempo de espera de recuperación de transferencia que se aplique a todas las transferencias para un agente de origen añadiendo un parámetro de tiempo de espera de recuperación de transferencia al archivo `agent.properties` del agente. También puede establecer un tiempo de espera de recuperación de transferencia para una transferencia individual desde la línea de mandatos o con IBM MQ Explorer o utilizando las tareas de Apache Ant. Si hay un valor de tiempo de espera de recuperación de transferencia establecido en el archivo `agent.properties`, establecer el tiempo de espera de recuperación de transferencia para una transferencia individual altera temporalmente el valor del archivo `agent.properties`.

Hay tres opciones para el tiempo de espera de recuperación de transferencia:

- El agente continúa intentando recuperar la transferencia estancada hasta que se completa correctamente. Este comportamiento es el mismo que el comportamiento predeterminado del agente si el tiempo de espera de recuperación de transferencia no está establecido.
- El agente marca la transferencia como fallada inmediatamente al entrar en la recuperación.
- El agente sigue reintentando la transferencia estancada durante el tiempo especificado antes de que la transferencia se marque como fallada.

Es opcional establecer el tiempo de espera de recuperación de transferencia. Si no lo establece, las transferencias siguen el comportamiento predeterminado. Esto es lo mismo que el comportamiento predeterminado de un agente de origen de Managed File Transfer antes de IBM MQ 9.1, donde el agente sigue intentando recuperar una transferencia bloqueada hasta que sea satisfactoria.

### Conceptos relacionados

“Recuperación y reinicio de MFT” en la página 314

Si el agente o el gestor de colas están no disponibles por algún motivo, por ejemplo, debido a un fallo en el suministro o en la red, Managed File Transfer se recupera tal como se indica a continuación en los siguientes escenarios:

## V 9.1.0 Conceptos de tiempo de espera de recuperación de transferencia

Puede establecer la cantidad de tiempo, en segundos, durante el cual un agente de origen sigue intentando recuperar una transferencia de archivo estancada. Si la transferencia no resulta satisfactoria cuando el agente alcanza el tiempo de espera para el intervalo de reintento, falla la transacción.

### Prioridad de tiempo de espera de recuperación

El valor de tiempo de espera de recuperación de transferencia para una transferencia individual especificado a través de los mandatos **fteCreateTransfer**, **fteCreateTemplate** o **fteCreateMonitor**, o utilizando IBM MQ Explorer, o tal como se ha especificado en el elemento anidado **fte:filespec**, tiene prioridad sobre el valor que se ha especificado para el parámetro **transferRecoveryTimeout** en el archivo `agent.properties` para el agente de origen.

Por ejemplo, si se inicia el mandato **fteCreateTransfer** sin el parámetro **-rt** y el par de valor, el agente de origen AGENT1 comprueba en el archivo `agent.properties` el valor de **transferRecoveryTimeout** con el fin de determinar el comportamiento del tiempo de espera de recuperación:

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -df C:\import\transferredfile.txt  
C:\export\originalfile.txt
```

Si el parámetro **transferRecoveryTimeout** del archivo `agent.properties` no se ha establecido o se ha establecido en `-1`, el agente sigue el comportamiento predeterminado e intenta recuperar la transferencia hasta que se lleve a cabo de forma satisfactoria.

Sin embargo, si el mandato **fteCreateTransfer** incluye el parámetro **-rt**, el valor de este parámetro tiene prioridad sobre el valor del archivo `agent.properties` y se utiliza como valor de tiempo de espera de recuperación para la transferencia:

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -rt 21600 -df C:\import\transferredfile.txt  
C:\export\originalfile.txt
```

### Contador de tiempo de espera de recuperación

El contador del tiempo de espera de recuperación se inicia cuando la transferencia entra en estado de recuperación. Se publica un mensaje de registro de transferencia en SYSTEM.FTE con la serie de tema `Log/agent_name/transfer_ID` para indicar que el estado de la transferencia ha cambiado a recuperación y la hora del reloj del agente de origen a la que ha cambiado el estado. Si la transferencia se reanuda dentro del intervalo de reintento y no alcanza el tiempo de espera de recuperación (`counter <= recovery timeout`), se restablecerá el contador en 0, listo para volver a iniciarse si la transferencia entra en fase de recuperación.

Si el contador alcanza el valor máximo establecido para el tiempo de espera de recuperación (`counter == recovery timeout`), la recuperación de la transferencia se detendrá y el agente de origen informará de que la transferencia ha fallado. Este tipo de anomalía de transferencia, provocada por el hecho de que la transferencia ha alcanzado el tiempo de espera de recuperación, se indica mediante el código de mensaje, `RECOVERY TIMEOUT (69)`. Se publica otro mensaje de registro de transferencias en el tema SYSTEM.FTE, con una serie de tema de `Log/agent_name/transfer_ID`, para indicar que ha

fallado la transferencia e incluye un mensaje, el código de retorno y el registro de sucesos del agente de origen. El registro de sucesos del agente de origen se actualiza con un mensaje cuando se produce alguna de las situaciones siguientes durante la recuperación:

- Cuando el parámetro de tiempo de espera de recuperación se establece en un valor superior a -1, la transferencia entra en fase de recuperación. El registro de sucesos del agente se actualiza para indicar el inicio del temporizador de la recuperación para **TransferId** y la cantidad de tiempo que espera el agente de origen hasta que inicia el proceso de tiempo de espera de recuperación.
- Cuando se reanuda la transferencia, se actualiza el registro de sucesos del agente de origen con un mensaje nuevo para indicar que se ha reanudado **TransferId** que estaba en recuperación.
- Cuando se agota el tiempo de espera de recuperación de transferencia, el registro de sucesos del agente de origen se actualiza para indicar que ha fallado **TransferId** durante la recuperación debido al tiempo de espera de recuperación.

Estos mensajes de registro permiten que los usuarios (suscriptores y registradores) identifiquen las transferencias que han fallado debido al tiempo de espera de recuperación de transferencias.

El contador del tiempo de espera de recuperación siempre está en el agente de origen. Sin embargo, si el agente de destino no consigue recibir información del agente de origen de forma puntual, puede enviar una solicitud al agente de origen para que ponga la transferencia en fase de recuperación. Para una transferencia cuya opción de tiempo de espera de recuperación se ha establecido, el agente de origen inicia el contador del tiempo de espera de recuperación cuando recibe la solicitud del agente de destino.

Sigue necesitándose una gestión manual para las transferencias que no utilizan la opción de tiempo de espera de recuperación, las transferencias fallidas y las que se han completado parcialmente.

Para conjuntos de transferencias cuya solicitud de transferencia única se emite para varios archivos, y algunos archivos se han completado correctamente pero uno se ha completado parcialmente, la transferencia seguirá marcada como fallida porque no se ha completado según lo previsto. Es posible que el agente de origen haya agotado el tiempo de espera durante la transferencia parcial del archivo.

Asegúrese de que el agente de destino y el servidor de archivos estén listos y que tengan un estado para aceptar transferencias de archivos.

Debe emitir de nuevo la solicitud de transferencia para todo el conjunto pero para evitar problemas debido a que algunos archivos permanecen en el destino desde el intento inicial de transferencia, puede emitir la nueva solicitud especificando la opción `sobrescribir si existe`. De este modo se asegura de que se borre el conjunto de archivos incompletos del intento de transferencia anterior como parte de la transferencia nueva, antes de grabar de nuevo los archivos en el destino.

**V 9.1.5** A partir de IBM MQ 9.1.5, ya no es necesario eliminar manualmente los archivos de partes que quedan en un destino después de que un intento de transferencia inicial haya fallado. Si se establece un tiempo de espera de recuperación de transferencia para una transferencia, el agente de origen cambia la transferencia al estado `RecoveryTimedOut` si se excede el tiempo de espera de la recuperación de transferencia. Después de que se haya resincronizado la transferencia, el agente de destino elimina los archivos de partes que se crearon durante la transferencia y envía un mensaje de finalización al agente de origen.

## Rastreo y mensajes

Los puntos de rastreo se incluyen con fines de diagnóstico. Se registra el valor de tiempo de espera de recuperación, el inicio del intervalo de reintentos, el inicio del periodo de reanudación y el restablecimiento del contador, así como si se ha agotado el tiempo de espera de la transferencia y si ha resultado fallida. En el caso de que hubiera problemas o un comportamiento inesperado, puede recopilar el registro de salida del agente de origen y los archivos de rastreo, y proporcionarlos cuando se lo solicite el personal de soporte de IBM para ayudarle a resolver problemas.

Los mensajes le avisan cuando:

- Una transferencia entra en recuperación ([BFGTR0081I](#))

- Una transferencia ha terminado porque ha excedido el tiempo de espera de la recuperación (BFGSS0081E)
- Se reanuda una transferencia después de haber estado en recuperación (BFGTR0082I)

### Conceptos relacionados

“Recuperación y reinicio de MFT” en la página 314

Si el agente o el gestor de colas están no disponibles por algún motivo, por ejemplo, debido a un fallo en el suministro o en la red, Managed File Transfer se recupera tal como se indica a continuación en los siguientes escenarios:

## V 9.1.0 Establecimiento del tiempo de espera de recuperación de transferencia para todas las transferencias de un agente de origen

Puede establecer un tiempo de espera de recuperación de transferencia que se aplique a todas las transferencias de un agente de origen añadiendo el parámetro **transferRecoveryTimeout** al archivo `agent.properties`.

### Acerca de esta tarea

Para establecer un tiempo de espera de recuperación de transferencia que se aplique a todas las transferencias de un agente de origen, añada el parámetro y valor para **transferRecoveryTimeout** al archivo `agent.properties`.

Hay tres opciones para el parámetro **transferRecoveryTimeout**:

**-1**

El agente sigue intentando recuperar la transferencia estancada hasta que ésta resulta satisfactoria. La utilización de esta opción equivale al comportamiento predeterminado del agente cuando la propiedad no se ha establecido.

**0**

El agente detiene la transferencia de archivo tan pronto como se inicia la recuperación.

**>0**

El agente sigue intentando recuperar la transferencia estancada durante el periodo de tiempo en segundos según se haya establecido mediante el valor entero positivo especificado.

Los cambios que realice en el archivo `agent.properties` entrarán en vigor sólo después de reiniciar el agente.

Si es necesario, puede alterar temporalmente el valor de tiempo de espera de recuperación de transferencia en el archivo `agent.properties` para una transferencia individual. Para obtener más información, consulte “Establecimiento del tiempo de espera de recuperación de transferencia para transferencias individuales” en la página 319.

### Procedimiento

- Para especificar que el agente sigue intentando recuperar la transferencia estancada hasta que se complete correctamente, establezca un valor de tiempo de espera de recuperación de transferencia de `-1`, tal como se muestra en el ejemplo siguiente:

```
transferRecoveryTimeout=-1
```

- Para especificar que el agente marca la transferencia como fallida inmediatamente después de entrar en la recuperación, establezca un valor de tiempo de espera de recuperación de transferencia de `0`, tal como se muestra en el ejemplo siguiente:

```
transferRecoveryTimeout=0
```

- Para especificar que el agente sigue reintentando una transferencia estancada durante una cantidad determinada de tiempo antes de que la transferencia se marque como fallida, establezca un valor de

tiempo de espera de recuperación de transferencia durante el periodo de tiempo, en segundos, que desea que el agente siga reintentando.

Por ejemplo, si se establece un valor de tiempo de espera de recuperación de transferencia en 21600 significa que el agente sigue intentando recuperar la transferencia durante seis horas desde el momento en que entra en recuperación:

```
transferRecoveryTimeout=21600
```

El valor máximo para este parámetro es 999999999.

## **V 9.1.0** Establecimiento del tiempo de espera de recuperación de transferencia para transferencias individuales

Puede establecer un tiempo de espera de recuperación de transferencia para una transferencia individual desde la línea de mandatos o con IBM MQ Explorer o utilizando las tareas de Apache Ant. Si hay un valor de tiempo de espera de recuperación de transferencia establecido en el archivo `agent.properties`, establecer el tiempo de espera de recuperación de transferencia para una transferencia individual altera temporalmente el valor establecido en el archivo `agent.properties`.

### **Acerca de esta tarea**

Puede establecer el parámetro de tiempo de espera de recuperación de transferencia para una transferencia individual cuando:

- Esté creando una transferencia utilizando el mandato **fteCreateTransfer** o utilizando IBM MQ Explorer.
- Esté creando una plantilla de transferencia utilizando el mandato **fteCreateTemplate** o utilizando IBM MQ Explorer.
- Esté creando un supervisor de recursos utilizando el mandato **fteCreateMonitor** o utilizando IBM MQ Explorer.
- Copiar o mover archivos utilizando las tareas `fte: filecopy` o `fte: filemove` Ant .

Si establece un valor de tiempo de espera de recuperación de transferencia para una transferencia individual, este valor altera temporalmente el valor de tiempo de espera de recuperación de transferencia establecido en el archivo `agent.properties` (consulte “Establecimiento del tiempo de espera de recuperación de transferencia para todas las transferencias de un agente de origen” en la página 318).

### **Procedimiento**

- Para utilizar el mandato **fteCreateTransfer** o **fteCreateTemplate** para establecer el tiempo de espera de recuperación de transferencia, especifique la opción adecuada para el parámetro **-rt**:
  - 1**  
El agente sigue intentando recuperar la transferencia estancada hasta que ésta resulta satisfactoria. La utilización de esta opción equivale al comportamiento predeterminado del agente cuando la propiedad no se ha establecido.
  - 0**  
El agente detiene la transferencia de archivo tan pronto como se inicia la recuperación.
  - >0**  
El agente sigue intentando recuperar la transferencia estancada durante el periodo de tiempo en segundos.

Ejemplos del mandato **fteCreateTransfer**

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -rt -1 -df C:\import\transferredfile.txt  
C:\export\originalfile.txt
```

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -rt 0 -df C:\import\transferredfile.txt
C:\export\originalfile.txt
```

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -rt 21600 -df C:\import\transferredfile.txt
C:\export\originalfile.txt
```

## Ejemplos del mandato **fteCreateTemplate**

```
fteCreateTemplate -tn "payroll accounts monthly report template" -rt -1 -sa PAYROLL -sm
QM_PAYROLL1 -da ACCOUNTS
-dm QM_ACCOUNTS -df C:\payroll_reports\*.xls C:\out\*.xls
```

```
fteCreateTemplate -tn "payroll accounts monthly report template" -rt 0 -sa PAYROLL -sm
QM_PAYROLL1 -da ACCOUNTS
-dm QM_ACCOUNTS -df C:\payroll_reports\*.xls C:\out\*.xls
```

```
fteCreateTemplate -tn "payroll accounts monthly report template" -rt 21600 -sa PAYROLL -sm
QM_PAYROLL1 -da ACCOUNTS
-dm QM_ACCOUNTS -df C:\payroll_reports\*.xls C:\out\*.xls
```

No hay ningún parámetro **-rt** para el mandato **fteCreateMonitor**. Si establece el parámetro **-rt** con el mandato **fteCreateTransfer** y también establece el parámetro **-gt**, se incluirá el parámetro de tiempo de espera de recuperación en el documento XML con la definición de transferencia que se genera cuando se ejecuta el mandato **fteCreateTransfer**. A continuación, el supervisor de recursos utiliza este documento XML cuando ejecuta el mandato **fteCreateMonitor**. En el ejemplo siguiente, los detalles de tiempo de espera de recuperación de transferencia se incluirán en el archivo `task.xml`:

```
fteCreateMonitor -ma AgentName -md C:\mqmft\monitors -mn Monitor_Name -mt task.xml -tr
"fileSize>=5MB,*.zip"
```

- Para utilizar la página del asistente de Nueva transferencia, Nuevo supervisor o Nueva plantilla de IBM MQ Explorer para establecer el tiempo de espera de recuperación de transferencia, seleccione la opción necesaria en el campo **Tiempo de espera excedido de recuperación de transferencia** (segundos):

### Como agente de origen

Si selecciona **Como agente de origen**, se utiliza el valor del parámetro **transferRecoveryTimeout** del archivo `agent.properties` si está establecido, de lo contrario se aplica el comportamiento predeterminado para el tiempo de espera de recuperación de transferencia.

### Recuadro de lista numérico

Si especifica un tiempo en segundos en el recuadro de lista numérico, el agente continúa intentando recuperar la transferencia estancada durante el periodo de tiempo especificado.

### Ninguna

Si selecciona **None**, no se establece ningún tiempo de espera de recuperación de transferencia y el agente sigue intentando recuperar la transferencia estancada hasta que la transferencia resulta satisfactoria.

- Para establecer el tiempo de espera de recuperación utilizando tareas de Ant, incluir la opción **transferRecoveryTimeout** y el valor, con los elementos **fte:filecopy** o **fte:filemove** para mover o copiar archivos, por ejemplo:

### Ejemplo de **fte:filecopy**

```
<fte:filecopy cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
src="agent1@qm1" dst="agent2@qm2"
rcproperty="copy.result" transferRecoveryTimeout="0">
  <fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/file.bin"/>
</fte:filecopy>
```



## Ejemplo de **fte:filemove**

```
<fte:filemove cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  src=agent1@qm1 dst="agent2@qm2"
  rcproperty="move.result" transferRecoveryTimeout="21600">

  <fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/file.bin"/>
</fte:filemove>
```

Windows

Linux

AIX

## Administración de MQ Telemetry

MQ Telemetry se administra mediante IBM MQ Explorer o en una línea de mandatos. Utilice el explorador para configurar canales de telemetría, controlar el servicio de telemetría y supervisar los clientes MQTT que se conecten a IBM MQ. Configure la seguridad de MQ Telemetry utilizando JAAS, TLS y el gestor de autorizaciones sobre objetos de IBM MQ.

### Administración utilizando IBM MQ Explorer

Utilice el explorador para configurar canales de telemetría, controlar el servicio de telemetría y supervisar los clientes MQTT que se conecten a IBM MQ. Configure la seguridad de MQ Telemetry utilizando JAAS, TLS y el gestor de autorizaciones sobre objetos de IBM MQ.

### Administración utilizando la línea de mandatos

MQ Telemetry se puede administrar por completo en la línea de mandatos mediante los mandatos MQSC.

La documentación de MQ Telemetry también contiene scripts de ejemplo en los que se demuestra la utilización básica de la aplicación cliente de IBM MQ Telemetry Transport v3.

Lea y comprenda los ejemplos en Programas de ejemplo de IBM MQ Telemetry Transport antes de utilizarlos.

#### Conceptos relacionados

[MQ Telemetry](#)

#### Referencia relacionada

[Propiedades de MQXR](#)

Linux

AIX

## Configurar un gestor de colas para telemetría en Linux y

### AIX

Siga estos pasos manuales para configurar un gestor de colas que ejecute MQ Telemetry. Puede ejecutar un procedimiento automático para definir una configuración más simple utilizando el soporte de MQ Telemetry para IBM MQ Explorer.

### Antes de empezar

1. Consulte [Instalación de MQ Telemetry](#) para obtener información sobre cómo instalar IBM MQ y la característica MQ Telemetry.
2. Cree e inicie un gestor de colas. El gestor de colas se conoce como *qMgt* en esta tarea.
3. Como parte de esta tarea, debe configurar el servicio de telemetría (MQXR). Los valores de propiedad MQXR se almacenan en un archivo de propiedades específico de la plataforma: `mqxr_unix.properties`. Normalmente, no es necesario editar el archivo de propiedades MQXR directamente, ya que casi todos los valores se pueden configurar mediante mandatos de administración de MQSC o IBM MQ Explorer. Si decide editar directamente el archivo, detenga el gestor de colas antes de realizar los cambios. Consulte [Propiedades MQXR](#).

### Acerca de esta tarea

El soporte de MQ Telemetry para IBM MQ Explorer incluye un asistente y un procedimiento de mandato de ejemplo `sampleMQM`. Los usuarios establecen una configuración inicial utilizando el ID de usuario

invitado; consulte [Verificación de la instalación de MQ Telemetry utilizando IBM MQ Explorer y los programas de ejemplo de IBM MQ Telemetry Transport](#).

Siga los pasos de esta tarea para configurar MQ Telemetry manualmente utilizando diferentes esquemas de autorización.

## Procedimiento

1. Abra una ventana de mandatos en el directorio de ejemplos de telemetría.

El directorio de ejemplos de telemetría es `/opt/mqm/mqxr/samples`.

2. Cree la cola de transmisión de telemetría.

```
echo "DEFINE QLOCAL('SYSTEM.MQTT.TRANSMIT.QUEUE') USAGE(XMITQ) MAXDEPTH(100000)" | runmqsc qMgr
```

Cuando el servicio de telemetría (MQXR) se inicia por primera vez, se crea `SYSTEM.MQTT.TRANSMIT.QUEUE`.

En esta tarea se crea manualmente, porque `SYSTEM.MQTT.TRANSMIT.QUEUE` debe existir antes de que se inicie el servicio de telemetría (MQXR) para autorizar el acceso a la misma.

3. Establecer la cola de transmisión predeterminada

Cuando se inicia por primera vez el servicio de telemetría (MQXR), este no altera el gestor de colas para que `SYSTEM.MQTT.TRANSMIT.QUEUE` sea cola de transmisión predeterminada.

Para hacer que `SYSTEM.MQTT.TRANSMIT.QUEUE` sea la cola de transmisión predeterminada, debe alterar la propiedad de la cola de transmisión predeterminada. Modifique la propiedad utilizando IBM MQ Explorer o con el mandato en el ejemplo siguiente:

```
echo "ALTER QMGR DEFXMITQ('SYSTEM.MQTT.TRANSMIT.QUEUE')" | runmqsc qMgr
```

Al modificar la cola de transmisión predeterminada, se puede modificar también la configuración existente. La razón por la que se modifica la cola de transmisión predeterminada a `SYSTEM.MQTT.TRANSMIT.QUEUE` es facilitar el envío de mensajes directamente a los clientes de MQTT. Sin modificar la cola de transmisión predeterminada, debe añadir una definición de cola remota para cada cliente que reciba mensajes de IBM MQ Explorer; consulte [“Envío de un mensaje a un cliente directamente”](#) en la página 326.

4. Siga el procedimiento indicado en [“Autorizar a clientes MQTT a acceder a objetos de IBM MQ”](#) en la página 329 para crear uno o varios ID de usuario. Los ID de usuario tienen autorización para publicar, suscribirse y enviar publicaciones a clientes MQTT.
5. Instale el servicio de telemetría (MQXR)

```
cat /opt/<install_dir>/mqxr/samples/installMQXRService_unix.mqsc | runmqsc qMgr
```

Consulte también el código de ejemplo en [Figura 16 en la página 323](#).

6. Inicie el servicio.

```
echo "START SERVICE(SYSTEM.MQXR.SERVICE)" | runmqsc qMgr
```

El servicio de telemetría (MQXR) se inicia automáticamente cuando se inicia el gestor de colas.

En esta tarea se inicia manualmente, porque el gestor de colas ya se está ejecutando.

7. Utilizando IBM MQ Explorer, configure los canales de telemetría para que acepten las conexiones de clientes MQTT.

Los canales de telemetría deben estar configurados de tal forma que sus identidades sean uno de los ID definidos en el paso 4.

Consulte también [DEFINE CHANNEL \(MQTT\)](#).

8. Verifique la configuración ejecutando el cliente de ejemplo.

Para que el cliente de ejemplo funcione con el canal de telemetría, el canal debe autorizar al cliente a publicar, suscribirse y recibir publicaciones. El cliente de ejemplo se conecta al canal de telemetría en el puerto 1883 de forma predeterminada. Consulte también [Programas de ejemplo de IBM MQ Telemetry Transport](#).

## Ejemplo

La [Figura 16 en la página 323](#) muestra el mandato `runmqsc` para crear el `SYSTEM.MQXR.SERVICE` manualmente en Linux.

```
DEF SERVICE(SYSTEM.MQXR.SERVICE) +
CONTROL(QMGR) +
DESCR('Manages clients using MQXR protocols such as MQTT') +
SERVTYPE(SERVER) +
STARTCMD('+MQ_INSTALL_PATH+/mqxr/bin/runMQXRService.sh') +
STARTARG('-m +QMNAME+ -d "+MQ_Q_MGR_DATA_PATH+" -g "+MQ_DATA_PATH+"') +
STOPCMD('+MQ_INSTALL_PATH+/mqxr/bin/endMQXRService.sh') +
STOPARG('-m +QMNAME+') +
STDOUT('+MQ_Q_MGR_DATA_PATH+/mqxr.stdout') +
STDERR('+MQ_Q_MGR_DATA_PATH+/mqxr.stderr')
```

Figura 16. `installMQXRService_unix.mqsc`

## Windows

## Configurar un gestor de colas para telemetría en Windows

Siga estos pasos manuales para configurar un gestor de colas que ejecute MQ Telemetry. Puede ejecutar un procedimiento automático para definir una configuración más simple utilizando el soporte de MQ Telemetry para IBM MQ Explorer.

### Antes de empezar

1. Consulte [Instalación de MQ Telemetry](#) para obtener información sobre cómo instalar IBM MQ y la característica MQ Telemetry.
2. Cree e inicie un gestor de colas. El gestor de colas se conoce como `qMgr` en esta tarea.
3. Como parte de esta tarea, debe configurar el servicio de telemetría (MQXR). Los valores de propiedad MQXR se almacenan en un archivo de propiedades específico de la plataforma: `mqxr_win.properties`. Normalmente, no es necesario editar el archivo de propiedades MQXR directamente, ya que casi todos los valores se pueden configurar mediante mandatos de administración de MQSC o IBM MQ Explorer. Si decide editar directamente el archivo, detenga el gestor de colas antes de realizar los cambios. Consulte [Propiedades MQXR](#).

### Acerca de esta tarea

El soporte de MQ Telemetry para IBM MQ Explorer incluye un asistente y un procedimiento de mandato de ejemplo `sampleMQM`. Los usuarios establecen una configuración inicial utilizando el ID de usuario invitado; consulte [Verificación de la instalación de MQ Telemetry utilizando IBM MQ Explorer](#) y los programas de ejemplo de [IBM MQ Telemetry Transport](#).

Siga los pasos de esta tarea para configurar MQ Telemetry manualmente utilizando diferentes esquemas de autorización.

### Procedimiento

1. Abra una ventana de mandatos en el directorio de ejemplos de telemetría.  
El directorio de ejemplos de telemetría es `WMQ program installation directory\mqxr\samples`.
2. Cree la cola de transmisión de telemetría.

```
echo DEFINE QLOCAL('SYSTEM.MQTT.TRANSMIT.QUEUE') USAGE(XMITQ) MAXDEPTH(100000) | runmqsc qMgr
```

Cuando el servicio de telemetría (MQXR) se inicia por primera vez, se crea SYSTEM.MQTT.TRANSMIT.QUEUE.

En esta tarea se crea manualmente, porque SYSTEM.MQTT.TRANSMIT.QUEUE debe existir antes de que se inicie el servicio de telemetría (MQXR) para autorizar el acceso a la misma.

3. Establecer la cola de transmisión predeterminada para qMgr

```
echo ALTER QMGR DEFQXMITQ('SYSTEM.MQTT.TRANSMIT.QUEUE') | runmqsc qMgr
```

*Figura 17. Establezca la cola de transmisión predeterminada*

Cuando se inicia por primera vez el servicio de telemetría (MQXR), este no altera el gestor de colas para que SYSTEM.MQTT.TRANSMIT.QUEUE sea cola de transmisión predeterminada.

Para hacer que SYSTEM.MQTT.TRANSMIT.QUEUE sea la cola de transmisión predeterminada, debe alterar la propiedad de la cola de transmisión predeterminada. Modifique la propiedad mediante IBM MQ Explorer o bien con el mandato que aparece en [Figura 17](#) en la [página 324](#).

Al modificar la cola de transmisión predeterminada, se puede modificar también la configuración existente. La razón por la que se modifica la cola de transmisión predeterminada a SYSTEM.MQTT.TRANSMIT.QUEUE es facilitar el envío de mensajes directamente a los clientes de MQTT. Sin modificar la cola de transmisión predeterminada, debe añadir una definición de cola remota para cada cliente que reciba mensajes de IBM MQ; consulte [“Envío de un mensaje a un cliente directamente”](#) en la [página 326](#).

4. Siga el procedimiento indicado en [“Autorizar a clientes MQTT a acceder a objetos de IBM MQ”](#) en la [página 329](#) para crear uno o varios ID de usuario. Los ID de usuario tienen autorización para publicar, suscribirse y enviar publicaciones a clientes MQTT.
5. Instale el servicio de telemetría (MQXR)

```
type  
installMQXRService_win.mqsc | runmqsc qMgr
```

6. Inicie el servicio.

```
echo START SERVICE(SYSTEM.MQXR.SERVICE) | runmqsc qMgr
```

El servicio de telemetría (MQXR) se inicia automáticamente cuando se inicia el gestor de colas.

En esta tarea se inicia manualmente, porque el gestor de colas ya se está ejecutando.

7. Utilizando IBM MQ Explorer, configure los canales de telemetría para que acepten las conexiones de clientes MQTT.

Los canales de telemetría deben estar configurados de tal forma que sus identidades sean uno de los ID definidos en el paso 4.

Consulte también [DEFINE CHANNEL \(MQTT\)](#).

8. Verifique la configuración ejecutando el cliente de ejemplo.

Para que el cliente de ejemplo funcione con el canal de telemetría, el canal debe autorizar al cliente a publicar, suscribirse y recibir publicaciones. El cliente de ejemplo se conecta al canal de telemetría en el puerto 1883 de forma predeterminada. Consulte también [Programas de ejemplo de IBM MQ Telemetry Transport](#).

### Creación manual de SYSTEM.MQXR.SERVICE

La [Figura 18](#) en la [página 325](#) muestra el mandato **runmqsc** para crear el SYSTEM.MQXR.SERVICE manualmente en Windows.

```

DEF SERVICE(SYSTEM.MQXR.SERVICE) +
CONTROL(QMGR) +
DESCR('Manages clients using MQXR protocols such as MQTT') +
SERVTYPE(SERVER) +
STARTCMD('+MQ_INSTALL_PATH+\mqxr\bin\runMQXRService.bat') +
STARTARG('-m +QMNAME+ -d "+MQ_Q_MGR_DATA_PATH+\." -g "+MQ_DATA_PATH+\."') +
STOPCMD('+MQ_INSTALL_PATH+\mqxr\bin\endMQXRService.bat') +
STOPARG('-m +QMNAME+') +
STDOUT('+MQ_Q_MGR_DATA_PATH+\mqxr.stdout') +
STDERR('+MQ_Q_MGR_DATA_PATH+\mqxr.stderr')

```

Figura 18. *installMQXRService\_win.mqsc*

## Windows Linux AIX Configuración de las colas distribuidas para enviar mensajes a clientes MQTT

Las aplicaciones de IBM MQ pueden enviar mensajes de clientes MQTT v3 publicando en una suscripción creada por un cliente, o enviando un mensaje directamente. Cualquiera que sea el método que se utilice, el mensaje se coloca en `SYSTEM.MQTT.TRANSMIT.QUEUE` y se envía al cliente mediante el servicio de telemetría (MQXR). Existen varias formas de colocar un mensaje en `SYSTEM.MQTT.TRANSMIT.QUEUE`.

### Publicación de un mensaje en respuesta a una suscripción de cliente MQTT

El servicio de telemetría (MQXR) crea una suscripción en nombre del cliente MQTT. El cliente es el destino de las publicaciones que coincidan con la suscripción que envíe el cliente. Los servicios de telemetría reenvían las publicaciones coincidentes al cliente.

Un cliente MQTT se conecta a IBM MQ como un gestor de colas, con su nombre de gestor de colas establecido en su `>IdentificadorCliente`. El destino de las publicaciones que se envían al cliente es una cola de transmisión, `SYSTEM.MQTT.TRANSMIT.QUEUE`. El servicio de telemetría reenvía mensajes en `SYSTEM.MQTT.TRANSMIT.QUEUE` a clientes MQTT, utilizando el nombre del gestor de colas de destino como clave para un cliente específico.

El servicio de telemetría (MQXR) abre la cola de transmisión utilizando `IdentificadorCliente` como el nombre del gestor de colas. El servicio de telemetría (MQXR) pasa el manejador de objeto de la cola a la llamada `MQSUB`, para reenviar las publicaciones que coincidan con la suscripción del cliente. En la resolución del nombre de objeto, se crea el `ClientIdentifier` como el nombre del gestor de colas remoto, y la cola de transmisión debe resolverse en `SYSTEM.MQTT.TRANSMIT.QUEUE`. Utilizando la resolución de nombres de objeto IBM MQ estándar, `ClientIdentifier` se resuelve de la forma siguiente; consulte [Tabla 21 en la página 326](#).

1. `ClientIdentifier` no coincide con nada.

`ClientIdentifier` es un nombre de gestor de colas remoto. No coincide con el nombre del gestor de colas local, con un alias de gestor de colas, ni con un nombre de cola de transmisión.

El nombre de cola no se ha definido. Actualmente, el servicio de telemetría (MQXR) establece `SYSTEM.MQTT.PUBLICATION.QUEUE` como el nombre de la cola. Un cliente MQTT v3 no da soporte a colas, por lo que el cliente ignora el nombre de cola que se ha resuelto.

El nombre de la propiedad del gestor de colas local, Cola de transmisión predeterminada, se debe establecer en `SYSTEM.MQTT.TRANSMIT.QUEUE`, de modo que la publicación se coloque en `SYSTEM.MQTT.TRANSMIT.QUEUE`, desde donde se enviará al cliente.

2. `ClientIdentifier` coincide con un alias de gestor de colas denominado `ClientIdentifier`.

`ClientIdentifier` es un nombre de gestor de colas remoto. Coincide con el nombre de un alias de gestor de colas.

El alias del gestor de colas debe definirse con `ClientIdentifier` como el nombre del gestor de colas remoto.

Al establecer el nombre de cola de transmisión en la definición de alias de gestor de colas, no es necesario que la transmisión predeterminada se establezca en `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Tabla 21. Resolución de nombres de un alias de gestor de colas MQTT					
	Entrada		Salida		
<i>ClientIdentifier</i>	Nombre del gestor de colas	Nombre de cola	Nombre del gestor de colas	Nombre de cola	Cola de transmisión
No coincide con nada	<i>ClientIdentifier</i>	<i>sin definir</i>	<i>ClientIdentifier</i>	<i>sin definir</i>	Cola de transmisión predeterminada.  SYSTEM.MQTT.TRANSMIT.QUEUE
Coincide con un alias de gestor de colas denominado <i>ClientIdentifier</i>	<i>ClientIdentifier</i>	<i>sin definir</i>	<i>ClientIdentifier</i>	<i>sin definir</i>	SYSTEM.MQTT.TRANSMIT.QUEUE

Para obtener más información sobre la resolución de nombres, consulte [Resolución de nombres](#).

Cualquier programa de IBM MQ puede publicar en el mismo tema. La publicación se envía a sus suscriptores, incluidos los clientes MQTT v3 que tengan una suscripción al tema.

Si se crea un tema administrativo en un clúster, con el atributo CLUSTER(*clusterName*), cualquier aplicación del clúster puede publicar en el cliente; por ejemplo:

```
echo DEFINE TOPIC('MQTTExamples') TOPICSTR('MQTT Examples') CLUSTER(MQTT) REPLACE | runmqsc qMgr
```

Figura 19. Definición de un tema de clúster en Windows

**Nota:** No proporcione a SYSTEM.MQTT.TRANSMIT.QUEUE un atributo de clúster.

Los suscriptores y publicadores de cliente MQTT pueden conectarse a gestores de colas distintos. Los suscriptores y publicadores pueden formar parte del mismo clúster, o pueden conectarse mediante una jerarquía de publicación/suscripción. La publicación se entrega del publicador al suscriptor utilizando IBM MQ.

## Envío de un mensaje a un cliente directamente

Una alternativa a que un cliente cree una suscripción y reciba una publicación que coincida con el tema de suscripción, es enviar un mensaje a un cliente MQTT v3 directamente. Las aplicaciones de cliente MQTT V3 no pueden enviar mensajes directamente, pero otras aplicaciones como, por ejemplo, las aplicaciones de IBM MQ, sí pueden.

La aplicación de IBM MQ debe conocer el `IdentificadorCliente` del cliente MQTT v3. Como los clientes MQTT v3 no tienen colas, el nombre de cola de destino se pasa al método MQTT v3 cliente de aplicaciones `messageArrived` como nombre de tema. Por ejemplo, en un programa MQI, cree un descriptor de objetos con el cliente como `ObjectQmgrName`:

```
MQOD.ObjectQmgrName = ClientIdentifier ;
MQOD.ObjectName = name ;
```

Figura 20. Descriptor de objetos MQI para enviar un mensaje a un destino de cliente MQTT v3

Si la aplicación se escribe utilizando JMS, cree un destino de punto a punto; por ejemplo:

```
javax.jms.Destination jmsDestination =
(javax.jms.Destination)jmsFactory.createQueue
("queue://ClientIdentifier/name");
```

Figura 21. Destino JMS para enviar un mensaje a un cliente MQTT v3

Para enviar un mensaje no solicitado a un cliente MQTT, utilice una definición de cola remota. El nombre del gestor de colas remoto debe resolverse en el `ClientIdentifier` del cliente. La cola de transmisión debe resolverse en `SYSTEM.MQTT.TRANSMIT.QUEUE`; consulte Tabla 22 en la página 327. El nombre de la cola remota puede ser cualquier cosa. El cliente lo recibe como una serie de tema.

Entrada		Salida		
Nombre de cola	Nombre del gestor de colas	Nombre de cola	Nombre del gestor de colas	Cola de transmisión
Nombre de la definición de la cola remota	En blanco o nombre de gestor de colas local	Nombre de cola remota utilizado como una serie de tema	<code>ClientIdentifier</code>	<code>SYSTEM.MQTT.TRANSMIT.QUEUE</code>

Si el cliente está conectado, el mensaje se envía directamente al cliente MQTT, que llama al método `messageArrived`; consulte [Método messageArrived](#).

Si el cliente se ha desconectado con una sesión persistente, el mensaje se almacena en `SYSTEM.MQTT.TRANSMIT.QUEUE`; consulte [Sesiones sin estado y con estado de MQTT](#). Se reenvía al cliente cuando éste se vuelve a conectar a la sesión.

Si envía un mensaje no persistente, éste se envía al cliente con una calidad de servicio de Como máximo una vez, `QoS=0`. Si envía un mensaje persistente directamente a un cliente, de forma predeterminada, se envía con la calidad de servicio exactamente una vez, `QoS=2`. Como es posible que el cliente no tenga un mecanismo de persistencia, el cliente puede reducir la calidad de servicio que acepta para los mensajes enviados directamente. Para reducir la calidad de servicio para los mensajes enviados directamente a un cliente, suscríbase al tema `DEFAULT.QoS`. Especifique la calidad de servicio máxima a la que el cliente puede dar soporte.

## Windows Linux AIX Identificación, autorización y autenticación de clientes MQTT

El servicio de telemetría (MQXR) publica temas de IBM MQ, o se suscribe a ellos, en nombre de clientes MQTT, utilizando canales MQTT. El administrador de IBM MQ configura la identidad de canal MQTT que se utiliza para la autorización de IBM MQ. El administrador debe definir una identidad común para el canal o utilizar el `Username` o `ClientIdentifier` de un cliente conectado al canal.

El servicio de telemetría (MQXR) puede autenticar el cliente utilizando el valor `Username` que proporciona el cliente o bien utilizando un certificado de cliente. El valor de `Username` se autentica utilizando una contraseña que proporciona el cliente.



En resumen: la identificación de cliente es la selección de la identidad de cliente. En función del contexto, el cliente se identifica con el valor `ClientIdentifier`, el valor `Username`, una identidad común de cliente que crea el administrador o un certificado de cliente. El identificador de cliente utilizado para la comprobación de la autenticidad no tiene que ser el mismo identificador que se utiliza para la autorización.

Los programas de cliente MQTT establecen el valor de `Username` y de `Password` que se envían al servidor utilizando un canal MQTT. También pueden establecer las propiedades TLS que se necesitan para cifrar y autenticar la conexión. El administrador decide si autenticar el canal MQTT y cómo autenticarlo.

Para autorizar a un cliente MQTT a acceder a objetos de IBM MQ, autorice el valor `ClientIdentifier`, o el valor `Username` del cliente, o autorice una identidad común del cliente. Para permitir que un cliente se conecte a IBM MQ, autentique el valor de `Username`, o utilice un certificado de cliente. Configure JAAS para autenticar el valor de `Username`, y configure TLS para autenticar un certificado de cliente.

Si define un valor `Password` en el cliente, cifre la conexión utilizando VPN, o configure el canal MQTT para que utilice TLS, para mantener la confidencialidad de la contraseña.

Resulta difícil gestionar certificados de clientes. Por este motivo, si los riesgos que conlleva la autenticación por contraseña resultan aceptables, ésta se utiliza a menudo para autenticar a los clientes.

Si existe una manera segura de gestionar y almacenar el certificado de cliente, se puede confiar en la autenticación con certificados. Sin embargo los certificados no suelen gestionarse de forma segura en los entornos en los que se utiliza la telemetría. En su lugar la autenticación de dispositivos que utilizan certificados de cliente se complementa con la autenticación de las contraseñas de cliente en el servidor. Debido a la complejidad adicional, la utilización de certificados de cliente se limita a aplicaciones altamente confidenciales. El uso de formas de autenticación se conoce como autenticación por dos factores. Debe conocer uno de los factores, por ejemplo, una contraseña, y tener otro, por ejemplo, un certificado.

En una aplicación en la que se necesite mucha confidencialidad como, por ejemplo, los dispositivos que utilicen la tecnología "chip and pin", el dispositivo se bloquea durante la fabricación para evitar intrusiones no autorizadas en el hardware y software internos. En el dispositivo se copia un certificado de cliente de confianza y duración limitada. El dispositivo se despliega en la ubicación en la que va a utilizarse. Cada vez que se utiliza el dispositivo, se realiza más autenticación, usando una contraseña u otro certificado de una tarjeta inteligente.

## **Identidad y autorización de cliente MQTT**

Utilice el ID de cliente, `Username`, o una identidad de cliente común para autorizar el acceso a objetos de IBM MQ.

El administrador de IBM MQ tiene tres opciones para seleccionar la identidad del canal MQTT. El administrador hace la elección al definir o modificar el canal MQTT que utiliza el cliente. La identidad se utiliza para autorizar el acceso a temas de IBM MQ. La elección se realiza en el orden siguiente:

1. El ID de cliente (consulte [USECLNTID](#)).
2. Una identidad que el administrador proporciona para el canal (el `MCAUSER` del canal). Consulte [MCAUSER](#)).
3. Si ninguna de las opciones anteriores es aplicable, el `Username` que se pasa desde el cliente MQTT (`Username` es un atributo de la clase `MqttConnectOptions`. Debe establecerse antes de que el cliente se conecte al servicio. Su valor predeterminado es nulo).

**Evitar problemas:** A la identidad elegida mediante este proceso se le hace referencia posteriormente en, por ejemplo, el mandato `DISPLAY CHSTATUS (MQTT)`, como el `MCAUSER` del cliente. Tenga en cuenta que esta no es necesariamente la misma identidad que el `MCAUSER` del canal al que se hace referencia en la opción (2).



Utilice el mandato de IBM MQ **setmqaut** para seleccionar qué objetos, y qué acciones, están autorizados para que los utilice la identidad asociada al canal MQTT. Por ejemplo, el siguiente código autoriza a la identidad del canal MQTTClient, proporcionada por el administrador del gestor de colas QM1:

```
setmqaut -m QM1 -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p MQTTClient -all +put
setmqaut -m QM1 -t topic -n SYSTEM.BASE.TOPIC -p MQTTClient -all +pub +sub
```

Windows

Linux

AIX

## Autorizar a clientes MQTT a acceder a objetos de IBM

### MQ

Siga estos pasos para autorizar a clientes MQTT a publicar y suscribirse a objetos de IBM MQ. Los pasos siguen cuatro patrones de control de acceso alternativos.

### Antes de empezar

A los clientes MQTT se les autoriza a acceder a objetos en IBM MQ asignándoles una identidad cuando se conectan a un canal de telemetría. El administrador de IBM MQ configura el canal de telemetría utilizando IBM MQ Explorer para asignar a un cliente uno de los tres tipos de identidad siguientes:

1. ClientIdentifier
2. Nombre de usuario
3. Un nombre que el administrador asigna al canal.

Sea cual sea el tipo que se utilice, la identidad debe estar definida en IBM MQ como principal por el servicio de autorización instalado. El servicio de autorización predeterminado en Windows o Linux se denomina OAM (Gestor de autorizaciones sobre objetos). Si va a utilizar el OAM, la identidad debe definirse como un ID de usuario.

Utilice la identidad para otorgar a un cliente, o grupo de clientes, permiso para publicar o suscribirse a temas definidos en IBM MQ. Si un cliente MQTT se ha suscrito a un tema, utilice la identidad para otorgarle permiso para recibir las publicaciones resultantes.

Resulta difícil gestionar un sistema con decenas de miles de clientes MQTT, ya que cada uno necesita permiso de acceso individual. Una solución es definir identidades comunes y asociar clientes MQTT individuales a una de las identidades comunes. Defina tantas identidades comunes como sea necesario para definir diferentes combinaciones de permisos. Otra solución es escribir un servicio de autorización propio que maneje los miles de usuarios de forma más sencilla a como lo hace el sistema operativo.

Puede combinar clientes MQTT en identidades comunes de dos formas, utilizando el OAM:

1. Defina varios canales de telemetría, cada uno con un ID de usuario diferente que el administrador asigna utilizando IBM MQ. Los clientes que se conectan utilizando diferentes números de puertos TCP/IP se asocian a diferentes canales de telemetría y se les asignan diferentes identidades.
2. Defina un único canal de telemetría, pero haga que cada cliente seleccione un valor de Username entre un conjunto pequeño de ID de usuario. El administrador configura el canal de telemetría para seleccionar el valor de Username del cliente como su identidad.

En esta tarea, la identidad del canal de telemetría se denomina *mqttUser*, independientemente de cómo se establezca. Si las colecciones de clientes utilizan identidades diferentes, utilice varios *mqttUsers*, uno para cada colección de clientes. Puesto que la tarea utiliza OAM, cada *mqttUser* debe ser un ID de usuario.

### Acerca de esta tarea

En esta tarea puede elegir entre cuatro patrones de control de acceso, que puede adaptar a requisitos específicos. Los patrones se diferencian en su granularidad del control de acceso.

- [“Sin control de acceso” en la página 330](#)
- [“Control de acceso de granularidad gruesa” en la página 330](#)
- [“Control de acceso de granularidad media” en la página 330](#)

- [“Control de acceso de granularidad precisa” en la página 330](#)

El resultado de los modelos es asignar a *mqttUsers* conjuntos de permisos para publicarlos y suscribirlos a IBM MQ y recibir publicaciones de IBM MQ.

#### *Sin control de acceso*

A los clientes MQTT se les asigna autorización administrativa de IBM MQ, y pueden realizar cualquier acción en cualquier objeto.

### Procedimiento

1. Cree un ID de usuario *mqttUser* para que actúe como la identidad de todos los clientes de MQTT.
2. Añada *mqttUser* al grupo *mqm*; consulte [Adición de un usuario a un grupo en Windows](#) o [Creación y gestión de grupos en Linux](#)

#### *Control de acceso de granularidad gruesa*

Los clientes MQTT tienen autorización para publicar y suscribirse, y para enviar mensajes a clientes MQTT. No tienen autorización para realizar otras acciones, ni para acceder a otros objetos.

### Procedimiento

1. Cree un ID de usuario *mqttUser* para que actúe como la identidad de todos los clientes de MQTT.
2. Autorice a *mqttUser* a publicar y suscribirse a todos los temas y a enviar publicaciones a los clientes de MQTT.

```
setmqaut -m qMgr -t topic -n SYSTEM.BASE.TOPIC -p mqttUser -all +pub +sub
setmqaut -m qMgr -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p mqttUser -all +put
```

#### *Control de acceso de granularidad media*

Los clientes MQTT se dividen en diferentes grupos para publicar y suscribirse a diferentes conjuntos de temas, y para enviar mensajes a clientes MQTT.

### Procedimiento

1. Cree varios ID de usuario *mqttUsers*, y varios temas administrativos en el árbol de temas de publicación/suscripción.
2. Autorice a diferentes *mqttUsers* para temas diferentes.

```
setmqaut -m qMgr -t topic -n topic1 -p mqttUserA -all +pub +sub
setmqaut -m qMgr -t topic -n topic2 -p mqttUserB -all +pub +sub
```

3. Cree un grupo *mqtt*, y añada todos los *mqttUsers* al grupo.
4. Autorice a *mqtt* a enviar temas a clientes de MQTT.

```
setmqaut -m qMgr -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p mqtt -all +put
```

#### *Control de acceso de granularidad precisa*

Los clientes MQTT se incorporan en un sistema de control de acceso existente que autoriza a los grupos a realizar acciones sobre objetos.

### Acerca de esta tarea

Se asigna un ID de usuario a uno o más grupos de sistema operativo en función de las autorizaciones que necesite. Si las aplicaciones de IBM MQ publican o se suscriben al mismo espacio de temas que los clientes MQTT, utilice este modelo. Se hace referencia a los grupos como Publish X, Subscribe Y y mqtt.

## Publish X

Los miembros de grupos de Publish X pueden publicar en *topicX*.

## Subscribe Y

Los miembros de los grupos Subscribe Y pueden suscribirse a *topicY*.

## mqtt

Los miembros del grupo *mqtt* pueden enviar publicaciones a clientes de MQTT.

## Procedimiento

1. Cree varios grupos, Publish X y Subscribe Y, que estén asignados a varios temas administrativos en el árbol de temas de publicación/suscripción.
2. Cree un grupo *mqtt*.
3. Cree varios ID de usuario *mqttUsers* y agregue los usuarios a cualquiera de los grupos, dependiendo qué estén autorizados a hacer.
4. Autorice distintos grupos Publish X y Subscribe X a distintos temas, y autorice al grupo *mqtt* para enviar mensajes a clientes MQTT .

```
setmqaut -m qMgr -t topic -n topic1 -p Publish X -all +pub
setmqaut -m qMgr -t topic -n topic1 -p Subscribe X -all +pub +sub
setmqaut -m qMgr -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p mqtt -all +put
```

Windows

Linux

AIX

## Autenticación de cliente MQTT mediante una contraseña

Autentique el valor de `Username` utilizando la contraseña del cliente. Puede autenticar el cliente utilizando una identidad diferente a la utilizada para autorizar al cliente a publicar temas y a suscribirse a los mismos.

El servicio de telemetría (MQXR) utiliza JAAS para autenticar el valor `Username` del cliente. JAAS utiliza la Contraseña proporcionada por el cliente MQTT.

El administrador de IBM MQ decide si se debe autenticar el valor de `Username`, o no autenticar nada, configurando el canal MQTT al que se conecta un cliente. Los clientes pueden asignarse a diferentes canales y cada canal puede configurarse para que autentique sus clientes de formas diferentes. Mediante JAAS, puede configurar qué métodos deben autenticar el cliente y cuáles pueden hacerlo de forma opcional.

La opción que elija para autenticar la identidad no afecta a la opción que elija para autorizar la identidad. Es posible que desee configurar una identidad común para la autorización para facilitar las tareas administrativas, pero que se autentique cada usuario para que utilice dicha identidad. En el procedimiento siguiente se describen los pasos para autenticar los usuarios individuales para que utilice una identidad común:

1. El administrador de IBM MQ establece la identidad de canal de MQTT en cualquier nombre, por ejemplo `MQTTClientUser`, utilizando IBM MQ Explorer.
2. El administrador de IBM MQ autoriza a `MQTTClient` publicar y suscribirse a cualquier tema:

```
setmqaut -m QM1 -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p MQTTClient -all +put
setmqaut -m QM1 -t topic -n SYSTEM.BASE.TOPIC -p MQTTClient -all +pub +sub
```

3. El desarrollador de aplicaciones cliente MQTT crea un objeto `MqttConnectOptions` y establece `Username` y `Password` antes de conectarse al servidor.
4. El desarrollador de seguridad crea un `LoginModule` JAAS para autenticar el valor `Username` con el valor `Password` y lo incluye en el archivo de configuración JAAS.
5. El administrador de IBM MQ configura el canal MQTT para autenticar el valor `UserName` del cliente utilizando JAAS.

Las conexiones entre el cliente MQTT y el gestor de colas las inicia siempre el cliente MQTT. El cliente MQTT es siempre el cliente SSL. Tanto la autenticación de cliente del servidor como la autenticación de servidor del cliente MQTT son opcionales.

Al proporcionar al cliente un certificado digital firmado privado, puede autenticar el cliente MQTT en WebSphere MQ. El administrador de WebSphere MQ puede forzar a los clientes de MQTT a autenticarse a sí mismos en el gestor de colas utilizando TLS. Sólo puede solicitar la autenticación de cliente como parte de una autenticación mutua.

Algunos tipos de redes privadas virtuales (VPN) como, por ejemplo, IPsec, autentican los puntos finales de una conexión TCP/IP, como alternativa a utilizar SSL. VPN cifra cada paquete IP que se transmite por la red. Una vez establecida la conexión VPN, se ha establecido la red de confianza. Puede conectar clientes MQTT a los canales de telemetría utilizando TCP/IP en la red VPN.

La autenticación de cliente mediante TLS depende de que el cliente tenga un secreto. Este dato secreto es la clave privada del cliente en el caso de un certificado autofirmado o una clave que proporciona una entidad emisora de certificados. La clave se utiliza para firmar el certificado digital del cliente. Todas las personas que tengan la correspondiente clave pública pueden verificar el certificado digital. Los certificados pueden ser de confianza o, si están en cadena, se les puede realizar un seguimiento a través de una cadena de certificados a un certificado raíz de confianza. La verificación de clientes envía todos los certificados de la cadena de certificados que proporciona el cliente al servidor. El servidor comprueba la cadena de certificados hasta que encuentra un certificado en el que confía. El certificado de confianza puede ser un certificado público generado a partir de un certificado autofirmado, o un certificado raíz que normalmente ha emitido una entidad emisora de certificados. Un último paso, que es opcional, es la comparación del certificado de confianza con una lista de revocación de certificados "activa".

El certificado de confianza puede haberlo emitido una entidad emisora de certificados y puede estar incluido en el almacén de certificados JRE. Puede ser un certificado autofirmado o cualquiera que se haya añadido al almacén de claves del canal de telemetría como un certificado de confianza.

**Nota:** El canal de telemetría tiene un almacén de claves/almacén de confianza combinado que contiene tanto las claves privadas de uno o varios canales de telemetría, y todos los certificados públicos necesarios para autenticar clientes. Puesto que un canal SSL debe tener un almacén de claves, y es el mismo archivo que el almacén de confianza, nunca se hace referencia al almacén de certificados JRE. La implicación es que si la autenticación de un cliente requiere un certificado raíz de CA, debe colocar el certificado raíz en el almacén de claves del canal, aunque el certificado raíz de CA ya esté en el almacén de certificados JRE. Nunca se hace referencia alguna al almacén de certificados JRE.

Considere las amenazas a las que la autenticación del cliente pretende hacer frente y qué papel juega el cliente y el servidor en esa situación. La autenticación del certificado de cliente sola no es suficiente para evitar accesos no autorizados al sistema. Si otra persona ha tomado posesión del dispositivo del cliente, éste no tiene necesariamente que estar actuando con la autoridad del titular del certificado. No confíe nunca en una única defensa frente a ataques no deseados. Utilice al menos dos factores para la autenticación, además de la posesión complementaria de un certificado con conocimiento de información privada. Por ejemplo, utilice JAAS y autentique el cliente utilizando una contraseña que emita el servidor.

La principal amenaza a la que se enfrenta un certificado de cliente es que caiga en las manos equivocadas. El certificado se encuentra en un almacén de claves protegido con contraseña en el cliente. ¿Cómo se coloca en el almacén de claves? ¿Cómo consigue el cliente MQTT la contraseña para el almacén de claves? ¿Qué nivel de seguridad tiene la protección con contraseña? Los dispositivos de telemetría a menudo son fáciles de eliminar y pueden ser pirateados en privado. ¿Debe el hardware del dispositivo estar protegido contra manipulación no autorizada? La distribución y protección de certificados del lado del cliente se reconoce como difícil; se denomina el problema de la gestión de claves.

Una amenaza secundaria es que el dispositivo se utilice de forma incorrecta para acceder a servidores de forma imprevista. Por ejemplo, si la aplicación MQTT está amenazada, es posible utilizar algún punto débil de la configuración del servidor mediante la identidad de cliente autenticada.

Para autenticar un cliente MQTT mediante SSL, configure el canal de telemetría y el cliente.

## Conceptos relacionados

[“Configuración de canal de telemetría para la autenticación de cliente MQTT mediante TLS” en la página 333](#)

El administrador de IBM MQ configura los canales de telemetría en el servidor. Cada canal se configura para aceptar una conexión TCP/IP en un número de puerto diferente. Los canales TLS se configuran con un acceso protegido con frase de contraseña a archivos de claves. Si un canal TLS está definido sin frase de contraseña o archivo de claves, el canal no acepta conexiones SSL.

## Tareas relacionadas

[Configuración de cliente MQTT para la autenticación de cliente mediante TLS](#)

## **Configuración de canal de telemetría para la autenticación de cliente MQTT mediante TLS**

El administrador de IBM MQ configura los canales de telemetría en el servidor. Cada canal se configura para aceptar una conexión TCP/IP en un número de puerto diferente. Los canales TLS se configuran con un acceso protegido con frase de contraseña a archivos de claves. Si un canal TLS está definido sin frase de contraseña o archivo de claves, el canal no acepta conexiones SSL.

Establezca la propiedad `com.ibm.mq.MQTT.ClientAuth` de un canal de telemetría TLS en `REQUIRED` para obligar a todos los clientes que se conectan en dicho canal a proporcionar una prueba de que tienen certificados digitales verificados. Los certificados de cliente se autentican utilizando certificados de las entidades emisoras de certificados, dirigidos a un certificado raíz de confianza. Si el certificado de cliente es autofirmado o está firmado por un certificado que proviene de una entidad emisora de certificados, los certificados firmados públicamente del cliente, o autoridad de certificados, deben almacenarse de forma segura en el servidor.

Coloque el certificado de cliente firmado públicamente o el certificado de la entidad emisora de certificados en el almacén de claves del canal de telemetría. En el servidor, los certificados firmados públicamente se almacenan en el mismo archivo de claves que los certificados firmados en privado, en vez de en un almacén de confianza aparte.

El servidor verifica la firma de los certificados de cliente que se envía utilizando todos los certificados públicos y paquetes de cifrado que tenga. El servidor verifica la cadena de claves. El gestor de colas puede configurarse para probar el certificado respecto a la lista de revocación de certificados. La propiedad de lista de nombres de revocación de gestor de colas es `SSLCRLNL`.

Si alguno de los certificados que un cliente envía lo verifica un certificado del almacén de claves del servidor, el cliente se autentica.

El administrador de IBM MQ puede configurar el mismo canal de telemetría para utilizar JAAS en la comprobación del valor `Username` o `ClientIdentifier` del cliente con el valor `Password` del cliente.

Puede utilizar el mismo almacén de claves para varios canales de telemetría.

La verificación de al menos un certificado digital en el almacén de claves de cliente protegido con contraseña en el dispositivo autentica al cliente en el servidor. El certificado digital sólo se utiliza para la autenticación mediante IBM MQ. No se utiliza para verificar la dirección TCP/IP del cliente, o establecer la identidad del cliente para la autorización o la contabilidad. La identidad del cliente adoptada por el servidor es `Username` o `ClientIdentifier` del cliente, o una identidad creada por el administrador de IBM MQ.

También puede utilizar las suites de cifrado TLS para la autenticación de cliente. Si piensa utilizar suites de cifrado SHA-2, consulte [“Requisitos del sistema para utilizar las suites de cifrado SHA-2 con canales de MQTT” en la página 335](#).

## Conceptos relacionados

[“Configuración del canal de telemetría para la autenticación de canal mediante TLS” en la página 334](#)

El administrador de IBM MQ configura los canales de telemetría en el servidor. Cada canal se configura para aceptar una conexión TCP/IP en un número de puerto diferente. Los canales TLS se configuran con un acceso protegido con frase de contraseña a archivos de claves. Si un canal TLS está definido sin frase de contraseña o archivo de claves, el canal no acepta conexiones SSL.

**Referencia relacionada**

[DEFINIR CANAL \(MQTT\)](#)

[ALTER CHANNEL \(MQTT\)](#)

Windows

Linux

AIX

## Autenticación de canal de telemetría mediante

### TLS

Las conexiones entre el cliente MQTT y el gestor de colas las inicia siempre el cliente MQTT. El cliente MQTT es siempre el cliente SSL. Tanto la autenticación de cliente del servidor como la autenticación de servidor del cliente MQTT son opcionales.

El cliente intenta siempre autenticar el servidor, a menos que esté configurado para utilizar una CipherSpec que dé soporte a la conexión anónima. Si la autenticación falla, la conexión no se establece.

Algunos tipos de redes privadas virtuales (VPN) como, por ejemplo, IPsec, autentican los puntos finales de una conexión TCP/IP, como alternativa a utilizar SSL. VPN cifra cada paquete IP que se transmite por la red. Una vez establecida la conexión VPN, se ha establecido la red de confianza. Puede conectar clientes MQTT a los canales de telemetría utilizando TCP/IP en la red VPN.

La autenticación de servidor mediante SSL autentica el servidor al que va a enviar información confidencial. El cliente realiza las comprobaciones que coinciden con los certificados enviados desde el servidor, con los certificados colocados en su almacén de confianza o en su almacén cacerts de JRE.

El almacén de certificados JRE es un archivo JKS, cacerts. Se encuentra en JRE InstallPath\lib\security\. Se instala con la contraseña predeterminada changeit. Puede almacenar los certificados en que confíe en el almacén de certificados de JRE, o en el almacén de confianza del cliente. No puede utilizar ambos almacenes. Utilice el almacén de confianza del cliente si desea mantener los certificados públicos en los que el cliente confía separados de los certificados que utilicen otras aplicaciones Java. Utilice el almacén de certificados JRE si desea utilizar un almacén de certificados común para todas las aplicaciones Java que ejecuten en el cliente. Si decide utilizar el almacén de certificados de JRE, revise los certificados que contenga, para asegurarse de que confía en ellos.

Puede modificar la configuración JSSE indicando un proveedor de confianza diferente. Puede personalizar un proveedor de confianza para que realice diferentes comprobaciones en un certificado. En algunos entornos OGSi que han utilizado el cliente MQTT, el entorno proporciona un proveedor de confianza diferente.

Para autenticar el canal de telemetría mediante TLS, configure el servidor y el cliente.

Windows

Linux

AIX

## Configuración del canal de telemetría para la autenticación de canal mediante TLS

El administrador de IBM MQ configura los canales de telemetría en el servidor. Cada canal se configura para aceptar una conexión TCP/IP en un número de puerto diferente. Los canales TLS se configuran con un acceso protegido con frase de contraseña a archivos de claves. Si un canal TLS está definido sin frase de contraseña o archivo de claves, el canal no acepta conexiones SSL.

Almacene el certificado digital en el servidor, firmado con su clave privada, en el almacén de claves que el canal de telemetría vaya a utilizar en el servidor. Almacene cualquier certificado de su cadena de claves en el almacén de claves, si desea transmitir la cadena de claves al cliente. Configure el canal de telemetría mediante IBM MQ Explorer para utilizar TLS. Proporcione la vía de acceso al almacén de claves y la frase de contraseña para acceder al mismo. Si no establece el número de puerto TCP/IP del canal, el número de puerto del canal de telemetría TLS toma, como valor predeterminado, el 8883.

También puede utilizar las suites de cifrado TLS para la autenticación de canal. Si piensa utilizar suites de cifrado SHA-2, consulte [“Requisitos del sistema para utilizar las suites de cifrado SHA-2 con canales de MQTT”](#) en la página 335.

## Conceptos relacionados

[“Configuración de canal de telemetría para la autenticación de cliente MQTT mediante TLS” en la página 333](#)

El administrador de IBM MQ configura los canales de telemetría en el servidor. Cada canal se configura para aceptar una conexión TCP/IP en un número de puerto diferente. Los canales TLS se configuran con un acceso protegido con frase de contraseña a archivos de claves. Si un canal TLS está definido sin frase de contraseña o archivo de claves, el canal no acepta conexiones SSL.

[CipherSpecs y CipherSuites](#)

## Referencia relacionada

[DEFINE CHANNEL \(MQTT\)](#)

[ALTER CHANNEL \(MQTT\)](#)

## Requisitos del sistema para utilizar las suites de cifrado SHA-2 con canales de MQTT

Si utiliza una versión de Java que sea compatible con las suites de cifrado SHA-2, puede utilizar estas suites para proteger los canales de MQTT (telemetría) y las aplicaciones cliente.

Para IBM MQ 8.0, que incluye el servicio de telemetría (MQXR), la versión mínima de Java es Java 7 de IBM, SR6. Las suites de cifrado SHA-2 están soportadas de forma predeterminada en Java 7 de IBM, SR4 y versiones posteriores. Por lo tanto, puede utilizar las suites de cifrado SHA-2 con el servicio de telemetría (MQXR) para proteger sus canales MQTT (telemetría).

Si está ejecutando un cliente de MQTT con un JRE diferente, asegúrese de que también sea compatible con las suites de cifrado SHA-2.

## Conceptos relacionados

[Servicio de telemetría \(MQXR\)](#)

[“Configuración del canal de telemetría para la autenticación de canal mediante TLS” en la página 334](#)

El administrador de IBM MQ configura los canales de telemetría en el servidor. Cada canal se configura para aceptar una conexión TCP/IP en un número de puerto diferente. Los canales TLS se configuran con un acceso protegido con frase de contraseña a archivos de claves. Si un canal TLS está definido sin frase de contraseña o archivo de claves, el canal no acepta conexiones SSL.

## Referencia relacionada

[DEFINIR CANAL \(MQTT\)](#)

[ALTER CHANNEL \(MQTT\)](#)

## Privacidad de las publicaciones en los canales de telemetría

La privacidad de las publicaciones MQTT enviadas en cualquier dirección en los canales de telemetría está protegida mediante TLS, para cifrar las transmisiones a través de la conexión.

Los clientes MQTT que se conectan a los canales de telemetría utilizan TLS para proteger la privacidad de las publicaciones transmitidas en el canal, mediante la criptografía de claves simétricas. Puesto que los puntos finales no se autentican, no se puede confiar en un canal de cifrado solo. Combine la protección de privacidad con la autenticación del servidor o mutua.

Algunos tipos de redes privadas virtuales (VPN) como, por ejemplo, IPsec, autentican los puntos finales de una conexión TCP/IP, como alternativa a utilizar SSL. VPN cifra cada paquete IP que se transmite por la red. Una vez establecida la conexión VPN, se ha establecido la red de confianza. Puede conectar clientes MQTT a los canales de telemetría utilizando TCP/IP en la red VPN.

Para obtener información sobre una configuración típica, que cifre el canal y autentique el servidor, consulte [“Autenticación de canal de telemetría mediante TLS” en la página 334](#).

El cifrado de las conexiones TLS sin autenticar el servidor expone la conexión frente a ataques de terceros. Aunque la información que se intercambia está protegida contra escuchas no autorizadas, no



sabe con quién se está intercambiando. A menos que controle la red, estará expuesto a que alguien intercepte las transmisiones IP, haciéndose pasar por el punto final.

Puede crear una conexión TLS cifrada, sin autenticar el servidor, mediante un intercambio de claves Diffie-Hellman CipherSpec que dé soporte a TLS anónima. El secreto maestro, compartido entre el cliente y el servidor, y que se utiliza para cifrar transmisiones TLS, se establece sin intercambiar ningún certificado de servidor firmado en privado.

Puesto que las conexiones anónimas son inseguras, la mayoría de las implementaciones TLS no toman como valor predeterminado la utilización de las CipherSpecs anónimas. Si un canal de telemetría acepta una conexión TLS que solicita un cliente, el canal deberá tener un almacén de claves protegido mediante una frase de contraseña. De forma predeterminada, puesto que las implementaciones TLS no utilizan las CipherSpecs anónimas, el almacén de claves debe contener un certificado de firma privada que demuestre que el cliente puede autenticarse.

Si utiliza las CipherSpecs anónima, el almacén de claves del servidor debe existir, pero no es necesario que contenga ningún certificado firmado en privado.

Otra forma de establecer una conexión cifrada es sustituir el proveedor de confianza en el cliente por su propia implementación. El proveedor de confianza no ha podría autenticar el certificado de servidor, pero la conexión se cifraría.



**Atención:** Al utilizar TLS con MQTT puede utilizar mensajes grandes, sin embargo, puede haber un posible impacto en el rendimiento al hacerlo. MQTT está optimizado para procesar mensajes pequeños (normalmente entre 1KB y 1MB de tamaño).

## Windows Linux AIX **Configuración TLS de clientes y canales de telemetría de MQTT Java**

Configure TLS para autenticar el canal de telemetría y el cliente de MQTT Java y cifrar la transferencia de mensajes entre ellos. MQTT Los clientes de Java utilizan JSSE (Java Secure Socket Extension) para conectar canales de telemetría utilizando TLS. Algunos tipos de redes privadas virtuales (VPN) como, por ejemplo, IPsec, autentican los puntos finales de una conexión TCP/IP, como alternativa a utilizar SSL. VPN cifra cada paquete IP que se transmite por la red. Una vez establecida la conexión VPN, se ha establecido la red de confianza. Puede conectar clientes MQTT a los canales de telemetría utilizando TCP/IP en la red VPN.

Puede configurar la conexión entre un cliente Java MQTT y un canal de telemetría para utilizar el protocolo TLS sobre TCP/IP. Lo que se asegura depende de cómo configure TLS para que utilice JSSE. Empezando con la configuración más segura, puede configurar tres niveles de seguridad diferentes:

1. Permita que se conecten sólo los clientes MQTT de confianza. Conecte un cliente MQTT sólo a un canal de telemetría de confianza. Cifre los mensajes entre el cliente y el gestor de colas; consulte [“Autenticación de cliente MQTT mediante TLS”](#) en la página 332
2. Conecte un cliente MQTT sólo a un canal de telemetría de confianza. Cifre los mensajes entre el cliente y el gestor de colas; consulte [“Autenticación de canal de telemetría mediante TLS”](#) en la página 334.
3. Cifre los mensajes entre el cliente y el gestor de colas; consulte [“Privacidad de las publicaciones en los canales de telemetría”](#) en la página 335.

### **Parámetros de configuración JSSE**

Modifique los parámetros JSSE para cambiar la forma en la que se configura una conexión TLS. Los parámetros de configuración de JSSE se organizan en tres conjuntos:

1. [Canal de MQ Telemetry](#)
2. [Cliente deMQTT Java](#)
3. [JRE](#)

Configure los parámetros del canal de telemetría mediante IBM MQ Explorer. Establezca los parámetros de cliente de MQTT Java en el atributo `MqttConnectionOptions.SSLProperties`. Modifique los



parámetros de seguridad JRE editando los archivos en el directorio de seguridad JRE en el cliente y en el servidor.

## Canal de MQ Telemetry

Establezca todos los parámetros TLS del canal de telemetría utilizando IBM MQ Explorer.

### ChannelName

ChannelName es un parámetro necesario en todos los canales.

El nombre del canal identifica el canal asociado a un número de puerto particular. Asignar un nombre a los canales le ayudará a administrar conjuntos de clientes MQTT.

### PortNumber

PortNumber es un parámetro opcional en todos los canales. El valor predeterminado es 1883 para canales TCP, y 8883 para los canales TLS.

El número de puerto TCP/IP asociado a este canal. Los clientes MQTT se conectan a un canal especificando el puerto definido para el canal. Si el canal tiene propiedades TLS, el cliente debe conectarse utilizando el protocolo TLS; por ejemplo:

```
MQTTClient mqttClient = new MqttClient( "ssl://www.example.org:8884", "clientId1");
mqttClient.connect();
```

### KeyFileName

KeyFileName es un parámetro necesario para los canales TLS. Con los canales TCP, debe omitirse.

KeyFileName es la vía de acceso al almacén de claves Java que contiene los certificados digitales que proporciona el usuario. Utilice JKS, JCEKS o PKCS12 como el tipo de almacén de claves en el servidor.

Identifique el tipo de almacén de claves utilizando una de las extensiones de archivo siguientes:

- .jks
- .jceks
- .p12
- .pkcs12

Un almacén de claves que tenga cualquier otra extensión de archivo se supone que es un almacén de claves JKS.

Puede combinar un tipo de almacén de claves en el servidor con otros tipos de almacén de claves en el cliente.

Coloque el certificado privado del servidor en el almacén de claves. El certificado se conoce como el certificado del servidor. El certificado puede estar autofirmado o formar parte de una cadena de certificados firmado por una autoridad de firmas.

Si está utilizando una cadena de certificados, coloque los certificados asociados en el almacén de claves del servidor.

El certificado del servidor y cualquier certificado de su cadena de certificados se envían a los clientes para autenticar la identidad del servidor.

Si ha establecido ClientAuth en Required, el almacén de claves debe contener los certificados necesarios para autenticar el cliente. El cliente envía un certificado autofirmado, o una cadena de certificados, y el cliente se autentica mediante la primera verificación de este material contra un certificado del almacén de claves. Con el uso de una cadena de certificados, un certificado puede verificar muchos clientes, incluso si se emiten con certificados de cliente distintos.

### PassPhrase

PassPhrase es un parámetro necesario para los canales TLS. Con los canales TCP, debe omitirse.

La frase de contraseña se utiliza para proteger el almacén de claves.

### **ClientAuth**

ClientAuth es un parámetro TLS opcional. Toma como valor predeterminado que no se autentique ningún cliente. Con los canales TCP, debe omitirse.

Establezca ClientAuth si desea que el servicio de telemetría (MQXR) autentique el cliente, antes de permitir que el cliente se conecte al canal de telemetría.

Si establece ClientAuth, el cliente debe conectarse al servidor utilizando TLS y autenticar el servidor. Como respuesta a la definición de ClientAuth, el cliente envía su certificado digital al servidor y cualquier otro certificado de su almacén de claves. El certificado digital se conoce como el certificado de cliente. Estos certificados se autentican con los certificados retenidos en el almacén de claves del canal y en el almacén de JRE cacerts.

### **CipherSuite**

CipherSuite es un parámetro TLS opcional. El valor predeterminado es intentar todas las CipherSpecs habilitadas. Con los canales TCP, debe omitirse.

Si desea utilizar una CipherSpec particular, establezca CipherSuite en el mismo nombre que la CipherSpec que debe utilizarse para establecer una conexión TLS.

El servicio de telemetría y el cliente MQTT negocian una CipherSpec común entre todas las CipherSpecs que están habilitadas en cada extremo. Si se especifica una CipherSpec determinada en uno de los extremos de la conexión, o en ambos, debe coincidir con la CipherSpec del otro extremo.

Instale otros cifrados añadiendo otros proveedores adicionales a JSSE.

### **Federal Information Processing Standards (FIPS)**

FIPS es un parámetro opcional. De forma predeterminada, no está definido.

Bien en el panel de propiedades del gestor de colas, o bien mediante **runmqsc**, establezca SSLFIPS. SSLFIPS especifica si sólo se deben utilizar algoritmos certificados por FIPS.

### **Revocation namelist**

Revocation namelist es un valor opcional. De forma predeterminada, no está definido.

Bien en el panel de propiedades del gestor de colas, o bien mediante **runmqsc**, establezca SSLCRLNL. SSLCRLNL especifica una lista de nombres de objetos de información de autenticación que se utilizan para proporcionar ubicaciones de revocación de certificados.

No se utiliza ningún otro parámetro de gestor de colas que defina propiedades TLS.

### **Cliente de MQTT Java**

Establezca propiedades TLS para el cliente Java en `MqttConnectionOptions.SSLProperties`; por ejemplo:

```
java.util.Properties sslClientProperties = new Properties();
sslClientProperties.setProperty("com.ibm.ssl.keyStoreType", "JKS");
com.ibm.micro.client.mqttv3.MqttConnectOptions conOptions = new MqttConnectOptions();
conOptions.setSSLProperties(sslClientProperties);
```

Los nombres y valores de propiedades específicas se describen en la clase `MqttConnectOptions`. Para obtener enlaces a la documentación de la API de cliente para las bibliotecas de cliente MQTT, consulte [Referencia de programación de cliente MQTT](#).

### **Protocol**

Protocol es opcional.

El protocolo se selecciona bajo negociación con el servidor de telemetría. Si necesita un protocolo específico, puede seleccionar uno. Si el servidor de telemetría no da soporte al protocolo, la conexión falla.

## ContextProvider

ContextProvider es opcional.

## KeyStore

KeyStore es opcional. Configúrelo si ha establecido ClientAuth en el servidor para forzar la autenticación del cliente.

Coloque el certificado digital del cliente, firmado mediante su clave privada, en el almacén de claves. Especifique la vía de acceso del almacén de claves y la contraseña. El tipo y el proveedor son opcionales. JKS es el tipo predeterminado, e IBMJCE el proveedor predeterminado.

Especifique un proveedor de almacén de claves diferente que haga referencia a una clase que añada un nuevo proveedor de almacén de claves. Pase el nombre del algoritmo utilizado por el proveedor del almacén de claves para crear una instancia de KeyManagerFactory estableciendo el nombre del gestor de claves.

## TrustStore

TrustStore es opcional. Puede colocar todos los certificados de confianza en el almacén de JREcacerts.

Configure el almacén de confianza si desea tener almacén de confianza diferente para el cliente. Es posible que no configure el almacén de confianza si el servidor está utilizando un certificado emitido por una CA conocida que ya tiene su certificado raíz almacenado en cacerts.

Añada el certificado firmado públicamente del servidor o el certificado raíz al almacén de confianza, y especifique la vía de acceso de almacén de confianza y la contraseña. JKS es el tipo predeterminado, e IBMJCE el proveedor predeterminado.

Especifique un proveedor de almacén de confianza diferente que haga referencia a una clase que añada un nuevo proveedor de almacén de confianza. Pase el nombre del algoritmo utilizado por el proveedor del almacén de confianza para crear una instancia de TrustManagerFactory estableciendo el nombre del gestor de confianza.

## JRE

En el JRE se configuran otros aspectos sobre la seguridad Java que afectan al comportamiento del TLS en el cliente y en el servidor. Los archivos de configuración en Windows están en *Java Installation Directory\jre\lib\security*. Si va a utilizar el JRE que se proporciona con IBM MQ la vía de acceso es la que se indica en la tabla siguiente:

Plataforma	Vía de acceso de archivo
Windows	<i>WMQ Installation Directory\java\jre\lib\security</i>
Plataformas UNIX and Linux	<i>WMQ Installation Directory/java/jre64/jre/lib/security</i>

## Entidades emisoras de certificados conocidas

El archivo cacerts contiene los certificados raíz de autoridades de certificación conocidas. cacerts se utiliza de forma predeterminada, a menos que especifique un almacén de confianza. Si utiliza el almacén de cacerts o no proporciona un almacén de confianza, debe revisar y editar la lista de firmantes en cacerts para cumplir los requisitos de seguridad.

Puede abrir cacerts utilizando el mandato IBM MQ `strmqikm` que ejecuta el programa de utilidad IBM Key Management. Abra cacerts como un archivo JKS, utilizando la contraseña `changeit`. Modifique la contraseña para proteger el archivo.

## Configuración de clases de seguridad

Utilice el archivo `java.security` para registrar proveedores de seguridad adicionales y otras propiedades de seguridad predeterminadas.

### Permisos

Utilice el archivo `java.policy` para modificar los permisos otorgados a los recursos. `javaws.policy` otorga permisos a `javaws.jar`

### Fuerza de cifrado

Algunos JRE se entregan con poca fuerza de cifrado. Si no puede importar claves a los almacenes de claves, la poca fuerza de cifrado puede ser la causa. Pruebe a iniciar **ikeyman** mediante el mandato **strmqikm**, o descargue archivos de política de jurisdicción de fuerza limitada desde [IBM Developer Kits, Security information](#).

**Importante:** Es posible que su país tenga restricciones sobre la importación, posesión, utilización y nueva exportación a otro país de software cifrado. Antes de descargar o utilizar archivos de políticas sin restricciones, debe comprobar las leyes existentes en su país. Compruebe sus regulaciones y políticas sobre importación, posesión, utilización y nueva exportación de software cifrado para ver si están permitidas estas acciones.

## Modificación del proveedor de confianza para permitir al cliente para conectarse a cualquier servidor

En el siguiente ejemplo se muestra cómo añadir un proveedor de confianza y cómo hacer referencia al mismo desde el código de cliente MQTT. En el ejemplo no se autentican el cliente ni el servidor. La conexión TLS resultante se cifra sin que se haya autenticado.

El fragmento de código de la Figura 22 en la página 340 establece el proveedor de confianza `AcceptAllProviders` y el gestor de confianza para el cliente MQTT.

```
java.security.Security.addProvider(new AcceptAllProvider());
java.util.Properties sslClientProperties = new Properties();
sslClientProperties.setProperty("com.ibm.ssl.trustManager", "TrustAllCertificates");
sslClientProperties.setProperty("com.ibm.ssl.trustStoreProvider", "AcceptAllProvider");
conOptions.setSSLProperties(sslClientProperties);
```

Figura 22. Fragmento de código de Cliente MQTT

```
package com.ibm.mq.id;
public class AcceptAllProvider extends java.security.Provider {
private static final long serialVersionUID = 1L;
public AcceptAllProvider() {
super("AcceptAllProvider", 1.0, "Trust all X509 certificates");
put("TrustManagerFactory.TrustAllCertificates",
AcceptAllTrustManagerFactory.class.getName());
}
}
```

Figura 23. `AcceptAllProvider.java`

```
protected static class AcceptAllTrustManagerFactory extends
javax.net.ssl.TrustManagerFactorySpi {
public AcceptAllTrustManagerFactory() {}
protected void engineInit(java.security.KeyStore keystore) {}
protected void engineInit(
javax.net.ssl.ManagerFactoryParameters parameters) {}
protected javax.net.ssl.TrustManager[] engineGetTrustManagers() {
return new javax.net.ssl.TrustManager[] { new AcceptAllX509TrustManager() };
}
}
```

Figura 24. `AcceptAllTrustManagerFactory.java`

```

protected static class AcceptAllX509TrustManager implements
javax.net.ssl.X509TrustManager {
public void checkClientTrusted(
java.security.cert.X509Certificate[] certificateChain,
String authType) throws java.security.cert.CertificateException {
report("Client authtype=" + authType);
for (java.security.cert.X509Certificate certificate : certificateChain) {
report("Accepting:" + certificate);
}
}
public void checkServerTrusted(
java.security.cert.X509Certificate[] certificateChain,
String authType) throws java.security.cert.CertificateException {
report("Server authtype=" + authType);
for (java.security.cert.X509Certificate certificate : certificateChain) {
report("Accepting:" + certificate);
}
}
public java.security.cert.X509Certificate[] getAcceptedIssuers() {
return new java.security.cert.X509Certificate[0];
}
private static void report(String string) {
System.out.println(string);
}
}

```

Figura 25. *AcceptAllX509TrustManager.java*

Windows

Linux

AIX

## Configuración JAAS del canal de telemetría

Configure JAAS para autenticar el valor de Username que envía el cliente.

El administrador de IBM MQ configura qué canales MQTT necesitan autenticación de cliente mediante JAAS. Especifique el nombre de una configuración JAAS para cada canal que vaya a realizar la autenticación JAAS. Los canales pueden utilizar todos la misma configuración JAAS o utilizar configuraciones JAAS diferentes. Las configuraciones se definen en *WMQData directory\qmgrs\qMgrName\mqxr\jaas.config*.

El archivo *jaas.config* está organizado por el nombre de configuración JAAS. Bajo cada nombre de configuración aparece una lista de las configuraciones de inicio de sesión; consulte la [“Ejemplo de archivo jaas.config”](#) en la página 342.

JAAS proporciona cuatro módulos de inicio de sesión estándar. Los módulos de inicio de sesión estándar de NT y UNIX son de valor limitado.

### JndiLoginModule

Autentica con un servicio de directorio configurado en JNDI (Java Naming and Directory Interface).

### Krb5LoginModule

Autentica utilizando protocolos Kerberos.

### NTLoginModule

Autentica utilizando la información de seguridad de NT para el usuario actual.

### UnixLoginModule

Autentica utilizando la información de seguridad de UNIX para el usuario actual.

El problema al utilizar *NTLoginModule* o *UnixLoginModule* es que el servicio de telemetría (MQXR) se ejecuta con la identidad *mqm*, y no con la identidad del canal MQTT. *mqm* es la identidad que se pasa a *NTLoginModule* o a *UnixLoginModule* para la autenticación, y no la identidad del cliente.

Para solucionar este problema, escriba un módulo de inicio de sesión propio o utilice otros módulos de inicio de sesión estándar. Se proporciona un *JAASLoginModule.java* de ejemplo con MQ Telemetry. Es una implementación de la interfaz *javax.security.auth.spi.LoginModule*. Utilícelo para desarrollar un método de autenticación propio.

Las clases nuevas de *LoginModule* que proporcione deben encontrarse en la vía de acceso de clase del servicio de telemetría (MQXR). No coloque sus clases en directorios de IBM MQ que estén en la vía de

acceso de clases. Cree sus propios directorios y defina la vía de acceso de clase completa del servicio de telemetría (MQXR).

Puede aumentar la vía de acceso de clases que utiliza el servicio de telemetría (MQXR) estableciendo la vía de acceso de clases en el archivo `service.env`. `CLASSPATH` debe estar en mayúsculas y la sentencia de vía de acceso de clase sólo puede contener literales. No puede utilizar variables en la `CLASSPATH`; por ejemplo, `CLASSPATH=%CLASSPATH%` es incorrecto. El servicio de telemetría (MQXR) establece su propia vía de acceso de clases. Se le añade `CLASSPATH` definida en `service.env`.

El servicio de telemetría (MQXR) proporciona dos devoluciones de llamada que devuelven el valor de `Username` y de `Password` del cliente conectado al canal MQTT. El `Nombre de usuario` y la `Contraseña` se establecen en el objeto `MqttConnectOptions`. Consulte [“Método JAASLoginModule.Login\(\) de ejemplo”](#) en la [página 342](#) para obtener un ejemplo de cómo acceder al `Nombre de usuario` y la `Contraseña`.

### Ejemplo de archivo `jaas.config`

Ejemplo de un archivo de configuración JAAS con una configuración con nombre, `MQXRConfig`.

```
MQXRConfig {
samples.JAASLoginModule required debug=true;
//com.ibm.security.auth.module.NTLoginModule required;
//com.ibm.security.auth.module.Krb5LoginModule required
//    principal=principal@your_realm
//    useDefaultCcache=TRUE
//    renewTGT=true;
//com.sun.security.auth.module.NTLoginModule required;
//com.sun.security.auth.module.UnixLoginModule required;
//com.sun.security.auth.module.Krb5LoginModule required
//    useTicketCache="true"
//    ticketCache="${user.home}/${}tickets";
};
```

### Método `JAASLoginModule.Login()` de ejemplo

Ejemplo de un módulo de inicio de sesión JAAS codificado para recibir el valor de `Username` y de `Password` que proporciona un cliente MQTT.

```
public boolean login()
throws javax.security.auth.login.LoginException {
    javax.security.auth.callback.Callback[] callbacks =
    new javax.security.auth.callback.Callback[2];
    callbacks[0] = new javax.security.auth.callback.NameCallback("NameCallback");
    callbacks[1] = new javax.security.auth.callback.PasswordCallback(
    "PasswordCallback", false);
    try {
        callbackHandler.handle(callbacks);
        String username = ((javax.security.auth.callback.NameCallback) callbacks[0])
        .getName();
        char[] password = ((javax.security.auth.callback.PasswordCallback) callbacks[1])
        .getPassword();
        // Accept everything.
        if (true) {
            loggedIn = true;
        } else
        {
            throw new javax.security.auth.login.FailedLoginException("Login failed");
        }

        principal= new JAASPrincipal(username);

    } catch (java.io.IOException exception) {
        throw new javax.security.auth.login.LoginException(exception.toString());
    } catch (javax.security.auth.callback.UnsupportedCallbackException exception) {
        throw new javax.security.auth.login.LoginException(exception.toString());
    }
}

return loggedIn;
}
```

### Tareas relacionadas

Resolución del problema: [El servicio de telemetría no ha llamado al módulo de inicio de sesión JAAS](#)

### Referencia relacionada

[Clase AuthCallback MQX](#)

## Administración de IBM MQ Light

Puede administrar MQ Light utilizando IBM MQ Explorer o en la línea de mandatos. Utilice Explorer para configurar canales y supervisar los clientes de MQ Light que están conectados a IBM MQ. Configure la seguridad de MQ Light utilizando TLS y JAAS.

### Antes de empezar

Para obtener información sobre la instalación de AMQP en la plataforma, consulte [Elección de qué instalar](#). Instale el componente MQP Service utilizando la renovación de fabricación IBM MQ V8.0.0.4 y no el fixpack V8.0.0.4. No puede instalar el componente AMQP en una versión del gestor de colas anterior a la versión 8.0.0.4.

### Administración utilizando IBM MQ Explorer

Utilice Explorer para configurar canales de AMQP y supervisar los clientes de MQ Light que están conectados a IBM MQ. Puede configurar la seguridad de MQ Light utilizando TLS y JAAS.

### Administración utilizando la línea de mandatos

Puede administrar MQ Light en la línea de mandatos [utilizando mandatos MQSC](#).

## Visualización de objetos de IBM MQ que utilizan clientes MQ Light

Puede ver los distintos recursos de IBM MQ que utilizan los clientes MQ Light, por ejemplo conexiones y suscripciones.

### Conexiones

Cuando se inicia el servicio AMQP se crean nuevos Hconns y se conectan al gestor de colas. Esta agrupación de Hconns se utiliza cuando los clientes MQ Light publican mensajes. Puede ver los Hconns utilizando el mandato **DISPLAY CONN**. Por ejemplo:

```
DISPLAY CONN(*) TYPE(CONN) WHERE (APPLDESC LK 'WebSphere MQ Advanced Message Queuing Protocol*')
```

Este mandato también muestra Hconns específicos del cliente. Los Hconns que tienen un atributo de ID de cliente en blanco son los Hconns que se utilizan en la agrupación

Cuando un cliente MQ Light se conecta a un canal AMQP, un nuevo Hconn se conecta al gestor de colas. Este Hconn se utiliza para consumir mensajes de forma asíncrona para las suscripciones que el cliente MQ Light ha creado. Puede ver el Hconn que utiliza un cliente MQ Light concreto mediante el mandato **DISPLAY CONN**. Por ejemplo:

```
DISPLAY CONN(*) TYPE(CONN) WHERE (CLIENTID EQ 'recv_abcd1234')
```

### Suscripciones creadas por clientes

Cuando un cliente MQ Light se suscribe a un tema, se crea una nueva suscripción de IBM MQ. El nombre de suscripción incluye la siguiente información:

- El nombre del cliente. Si el cliente se ha unido a una suscripción compartida, se utiliza el nombre de la unidad compartida.
- El patrón de tema al que se ha suscrito el cliente

- Un prefijo. El prefijo es `private` si el cliente ha creado una suscripción no compartida o `share` si el cliente se ha unido a una suscripción compartida

Para ver las suscripciones que utilizan un cliente MQ Light determinado, ejecute el mandato **DISPLAY SUB** y filtre por el prefijo `private`:

```
DISPLAY SUB(':private:*')
```

Para ver las suscripciones compartidas que clientes utilizan, ejecute el mandato **DISPLAY SUB** y filtre por el prefijo `share`:

```
DISPLAY SUB(':share:*')
```

Dado que las suscripciones compartidas pueden utilizarse por varios clientes MQ Light, puede que desee ver los clientes que actualmente consumen mensajes de la suscripción compartida. Puede hacerlo listando los Hconns que actualmente tienen un manejador abierto en la cola de suscripción. Para ver los clientes que actualmente utilizan una unidad compartida, realice los siguientes pasos:

1. Busque el nombre de cola que utiliza como destino la suscripción compartida. Por ejemplo:

```
DISPLAY SUB(':private:recv_e298452:public') DEST
5 : DISPLAY SUB(':private:recv_e298452:public') DEST
AMQ8096: WebSphere MQ subscription inquired.
SUBID(414D5120514D312020202020202020707E0A565C2D0020)
SUB(:private:recv_e298452:public)
DEST(SYSTEM.MANAGED.DURABLE.560A7E7020002D5B)
```

2. Ejecute el mandato **DISPLAY CONN** para buscar los manejadores abiertos en dicha cola:

```
DISPLAY CONN(*) TYPE(HANDLE) WHERE (OBJNAME
EQ SYSTEM.MANAGED.DURABLE.560A7E7020002D5B)
21 : DISPLAY CONN(*) TYPE(HANDLE) WHERE(OBJNAME EQ
SYSTEM.MANAGED.DURABLE.560A7E7020002D5B)

AMQ8276: Display Connection details.
CONN(707E0A56642B0020)
EXTCONN(414D5143514D312020202020202020)
TYPE(HANDLE)

OBJNAME(SYSTEM.BASE.TOPIC)          OBJTYPE(TOPIC)

OBJNAME(SYSTEM.MANAGED.DURABLE.560A7E7020002961)
OBJTYPE(Queue)
```

3. Para cada uno de los manejadores, vea el ID de cliente de MQ Light que tiene el manejador abierto:

```
DISPLAY CONN(707E0A56642B0020) CLIENTID
23 : DISPLAY CONN(707E0A56642B0020) CLIENTID

AMQ8276: Display Connection details.
CONN(707E0A56642B0020)
EXTCONN(414D5143514D312020202020202020)
TYPE(CONN)
CLIENTID(recv_8f02c9d)
DISPLAY CONN(707E0A565F290020) CLIENTID
24 : DISPLAY CONN(707E0A565F290020) CLIENTID
AMQ8276: Display Connection details.
CONN(707E0A565F290020)
EXTCONN(414D5143514D312020202020202020)
TYPE(CONN)
CLIENTID(recv_86d8888)
```

## Identificación, autorización y autenticación de clientes MQ Light

Como otras aplicaciones cliente IBM MQ, puede proteger las conexiones AMQP de varias formas.



Puede utilizar las siguientes características de seguridad para proteger las conexiones AMQP para IBM MQ:

- [Registros de autenticación de canal](#)
- [Autenticación de conexión](#)
- Configuración de usuario de MCA de canal
- Definiciones de autoridad de IBM MQ
- [Conectividad de TLS](#)

Desde una perspectiva de seguridad, establecer una conexión consta de los dos pasos siguientes:

- Decidir si la conexión debe continuar
- Decidir qué identidad de IBM MQ asume la aplicación para comprobaciones de autorización posteriores

La siguiente información describe distintas configuraciones de IBM MQ y los pasos que se siguen cuando un cliente AMQP intenta realizar una conexión. No todas las configuraciones de IBM MQ utilizan todos los pasos descritos. Por ejemplo, algunas configuraciones no utilizan TLS para conexiones dentro del cortafuegos de la empresa y algunas configuraciones utilizan TLS pero no utilizan certificados de cliente para la autenticación. Muchos entornos no utilizan módulos personalizados o JAAS personalizados.

## Establecer una conexión

Los siguientes pasos describen lo que sucede cuando se establece una conexión por un cliente AMQP. Los pasos determinan si la conexión continua y qué identidad de IBM MQ asume la aplicación para las comprobaciones de autorización:

1. Si el cliente abre una conexión TLS a IBM MQ y proporciona un certificado, el gestor de colas intenta validar el certificado de cliente.
2. Si el cliente proporciona el nombre de usuario y las credenciales de usuario, el gestor de colas recibe una trama AMQP SASL y se comprueba la configuración de MQ CONNAUTH.
3. Se comprueban las reglas de autenticación de canal MQ (por ejemplo, si la dirección IP y el DNS de certificado TLS son válidos)
4. El canal MCAUSER se confirma, a menos que las reglas de autenticación de canal determinen lo contrario.
5. Si se ha configurado un módulo JAAS, este se invoca
6. La comprobación de autorización de MQ CONNECT aplicada al ID de usuario MQ resultante.
7. Conexión establecida con una identidad de IBM MQ asumida.

## Publicar un mensaje

Los siguientes pasos describen lo que sucede cuando se publica un mensaje por un cliente AMQP. Los pasos determinan si la conexión continua y qué identidad de IBM MQ asume la aplicación para las comprobaciones de autorización:

1. La trama para adjuntar enlaces AMQP llega al gestor de colas. La autorización de publicación de IBM MQ para la serie de tema especificada se comprueba para la identidad de usuario MQ establecida durante la conexión.
2. El mensaje se publica en una serie de tema indicada.

## Suscribirse a un patrón de tema

Los siguientes pasos describen lo que sucede cuando un cliente AMQP se suscribe a un patrón de tema. Los pasos determinan si la conexión continua y qué identidad de IBM MQ asume la aplicación para las comprobaciones de autorización:

1. La trama para adjuntar enlaces AMQP llega al gestor de colas. La autorización de suscripción de IBM MQ para el patrón de tema especificado se comprueba para la identidad de usuario MQ establecida durante la conexión.
2. La suscripción se crea.

## Identidad y autorización de cliente MQ Light

Utilice el ID de cliente MQ Light, el nombre de usuario de MQ Light o una identidad de cliente común definida en el canal o en una regla de autenticación de canal, para obtener la autorización para acceder a objetos IBM MQ.

El administrador elige cuándo definir o modificar el canal AMQP, configurando el valor CONNAUTH del gestor de colas, o definiendo las reglas de autenticación de canal. La identidad se utiliza para autorizar el acceso a temas de IBM MQ. La elección se realiza basándose en lo siguiente:

1. El atributo USECLNTID del canal.
2. El atributo ADOPTCTX de la regla CONNAUTH del gestor de colas.
3. El atributo MCAUSER definido en el canal.
4. El atributo USERSRC de una regla de autenticación de canal coincidente.

**Evitar problemas:** Posteriormente, se hará referencia a la identidad que elija el proceso, por ejemplo mediante el mandato DISPLAY CHSTATUS (AMQP), como el MCAUSER del cliente. Tenga en cuenta que esta no es necesariamente la misma identidad que el MCAUSER del canal al que se hace referencia en la opción (2).

Utilice el mandato de IBM MQ **setmqaut** para seleccionar qué objetos, y qué acciones, están autorizados para que los utilice la identidad asociada al canal AMQP. Por ejemplo, los mandatos siguientes autorizan una identidad de canal AMQPClient, proporcionada por el administrador del gestor de colas QM1:

```
setmqaut -m QM1 -t topic -n SYSTEM.BASE.TOPIC -p AMQPClient -all +pub +sub
```

and

```
setmqaut -m QM1 -t qmgr -p AMQPClient -all +connect
```

## Autenticación de cliente MQ Light mediante una contraseña

Autentique el nombre de usuario MQ Light utilizando la contraseña de cliente. Puede autenticar el cliente utilizando una identidad diferente a la utilizada para autorizar al cliente a publicar temas y a suscribirse a los mismos.

El servicio AMQP puede utilizar MQ CONNAUTH o JAAS para autenticar el nombre de usuario del cliente. Si uno de ellos está configurado, la contraseña proporcionada por el cliente la verifica la configuración CONNAUTH MQ o el módulo JAAS.

El procedimiento siguiente describe los pasos de ejemplo para autenticar usuarios individuales con los usuarios y contraseñas del sistema operativo local y, si tiene éxito, adoptar la identidad común AMQPUser:

1. El administrador de IBM MQ establece la identidad MCAUSER del canal AMQP en cualquier nombre, como por ejemplo AMQPUser, utilizando IBM MQ Explorer.
2. El administrador de IBM MQ autoriza a AMQPUser a publicar y a suscribirse a cualquier tema:

```
setmqaut -m QM1 -t topic -n SYSTEM.BASE.TOPIC -p AMQPUser -all +pub +sub +connect
```

3. El administrador de IBM MQ configura una regla IDPWOS CONNAUTH para comprobar el nombre de usuario y la contraseña indicados por el cliente. La regla CONNAUTH debe establecer CHCKCLNT(REQUIRED) y ADOPTCTX(NO).

**Nota:** Se recomienda utilizar reglas de autenticación de canal y establecer el atributo de canal MCAUSER en un usuario que no tenga privilegios, para permitir un mayor control sobre las conexiones con el gestor de colas.

## Privacidad de las publicaciones en los canales

La privacidad de publicaciones AMQP enviadas a cualquier dirección mediante los canales AMQP se protege utilizando TLS para cifrar transmisiones a través de la conexión.

Los clientes AMQP que se conecten a los canales de AMQP utilizan TLS para proteger la privacidad de las publicaciones transmitidas en el canal, mediante el cifrado de claves simétricas. Puesto que los puntos finales no se autentican, no se puede confiar en un canal de cifrado solo. Combine la protección de privacidad con la autenticación del servidor o mutua.

Algunos tipos de redes privadas virtuales (VPN) como, por ejemplo, IPsec, autentican los puntos finales de una conexión TCP/IP, como alternativa a utilizar TLS. VPN cifra cada paquete IP que se transmite por la red. Una vez establecida la conexión VPN, se ha establecido la red de confianza. Puede conectar clientes AMQP a canales AMQP utilizando TCP/IP en la red VPN.

El cifrado de las conexiones TLS sin autenticar el servidor expone la conexión frente a ataques de terceros. Aunque la información que se intercambia está protegida contra escuchas no autorizadas, no sabe con quién se está intercambiando. A menos que controle la red, estará expuesto a que alguien intercepte las transmisiones IP, haciéndose pasar por el punto final.

Puede crear una conexión TLS cifrada, sin autenticar el servidor, mediante un intercambio de claves Diffie-Hellman CipherSpec que dé soporte a TLS anónima. El secreto maestro, compartido entre el cliente y el servidor, y que se utiliza para cifrar transmisiones TLS, se establece sin intercambiar ningún certificado de servidor firmado en privado.

Puesto que las conexiones anónimas son inseguras, la mayoría de las implementaciones TLS no toman como valor predeterminado la utilización de las CipherSpecs anónimas. Si un canal de AMQP acepta una conexión TLS que solicita un cliente, el canal deberá tener un almacén de claves protegido mediante una frase de contraseña. De forma predeterminada, puesto que las implementaciones TLS no utilizan las CipherSpecs anónimas, el almacén de claves debe contener un certificado de firma privada que demuestre que el cliente puede autenticarse.

Si utiliza las CipherSpecs anónima, el almacén de claves del servidor debe existir, pero no es necesario que contenga ningún certificado firmado en privado.

Otra forma de establecer una conexión cifrada es sustituir el proveedor de confianza en el cliente por su propia implementación. El proveedor de confianza no ha podría autenticar el certificado de servidor, pero la conexión se cifraría.

## Configuración de clientes MQ Light con TLS

Puede configurar clientes MQ Light para utilizar TLS para proteger flujo de datos en la red y para autenticar la identidad del gestor de colas al que se conecta el cliente.

Para utilizar TLS para la conexión de un cliente MQ Light a un canal AMQP, debe asegurarse de que el gestor de colas se ha configurado en TLS. [Configuración de TLS en los gestores de colas](#) describe cómo configurar el almacén de claves del que un gestor de colas lee los certificados TLS.

Cuando el gestor de colas se ha configurado con un almacén de claves, debe configurar los atributos TLS en el canal AMQP al que se conectarán los clientes. Los canales AMQP tienen siguientes cuatro atributos relacionados con la configuración de TLS:

### SSLCAUTH

El atributo SSLCAUTH se utiliza para especificar si el gestor de colas debe requerir un cliente MQ Light para presentar un certificado de cliente para verificar su identidad.

### SSLCIPH

El atributo SSLCIPH especifica el cifrado que el canal debe utilizar para codificar datos en el flujo TLS.

## SSLPEER

El atributo SSLPEER se utiliza para especificar el nombre distinguido (DN) con el que debe coincidir un certificado de cliente si una conexión va a estar permitida.

## CERTLABL

CERTLABL especifica el certificado que el gestor de colas debe presentar al cliente. El almacén de claves del gestor de colas puede contener varios certificados. Este atributo permite especificar el certificado que debe utilizarse para las conexiones con este canal. Si no se ha especificado ningún CERTLABL, se utiliza el certificado en el repositorio de claves del gestor de colas con la etiqueta que corresponde al atributo CERTLABL del gestor de colas.

Cuando haya configurado el canal AMQP con los atributos TLS, debe reiniciar el servicio AMQP utilizando el siguiente mandato:

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

Cuando un cliente MQ Light se conecta a un canal AMQP protegido por TLS, el cliente verifica la identidad del certificado presentado por el gestor de colas. Para ello debe configurar el cliente MQ Light con un almacén de confianza que contenga el certificado del gestor de colas. Los pasos para hacerlo varían en función del cliente MQ Light que esté utilizando.

- Para ver la documentación de la API de JS de Nodo del cliente MQ Light, consulte <https://www.npmjs.com/package/mqlight>
- Para ver la documentación de la API Java del cliente MQ Light, consulte <https://mqlight.github.io/java-mqlight/>
- Para ver la documentación de cliente MQ Light para Ruby, consulte <https://www.rubydoc.info/github/mqlight/ruby-mqlight/>
- Para ver la documentación de cliente MQ Light para Python, consulte <https://python-mqlight.readthedocs.org/en/latest/>

## Desconexión de clientes MQ Light del gestor de colas

Si desea desconectar MQ Light del gestor de colas, ejecute el mandato PURGE CHANNEL o detenga la conexión con el cliente MQ Light.

- Ejecute el mandato **PURGE CHANNEL**. Por ejemplo:

```
PURGE CHANNEL(MYAMQP) CLIENTID('recv_28dbb7e')
```

- Como alternativa, detenga la conexión que el cliente MQ Light está utilizando para desconectar el cliente realizando los siguientes pasos:

1. Busque la conexión que utiliza el cliente ejecutando el mandato **DISPLAY CONN**. Por ejemplo:

```
DISPLAY CONN(*) TYPE(CONN) WHERE (CLIENTID EQ 'recv_28dbb7e')
```

La salida del mandato es la siguiente:

```
DISPLAY CONN(*) TYPE(CONN) WHERE (CLIENTID EQ 'recv_28dbb7e')
40 : DISPLAY CONN(*) TYPE(CONN) WHERE (CLIENTID EQ 'recv_28dbb7e')
AMQ8276: Display Connection details.
CONN(707E0A565F2D0020)
EXTCONN(414D5143514D31202020202020202020)
TYPE(CONN)
CLIENTID(recv_28dbb7e)
```

2. Detenga la conexión. Por ejemplo:

```
STOP CONN(707E0A565F2D0020)
```

## Administración de multidifusión

Utilice esta información para conocer las tareas de administración de IBM MQ Multicast, tales como reducir el tamaño de los mensajes de multidifusión y habilitar la conversión de datos.

### Iniciación a la multidifusión

Utilice esta información para empezar a trabajar con temas y objetos de información de comunicación de IBM MQ Multicast.

#### Acerca de esta tarea

La mensajería de IBM MQ Multicast utiliza la red para entregar mensajes mediante la correlación de temas con direcciones de grupo. Las tareas siguientes son una forma rápida de probar si la dirección IP y puerto necesarios están configurados correctamente para la mensajería de multidifusión.

#### Creación de un objeto **COMMINFO** para multidifusión

El objeto de información de comunicación (COMMINFO) contiene los atributos asociados con la transmisión de multidifusión. Para obtener más información sobre los parámetros de objeto COMMINFO, consulte [DEFINE COMMINFO](#).

Utilice el siguiente ejemplo de línea de mandatos para definir un objeto COMMINFO para multidifusión:

```
DEFINE COMMINFO(MC1) GRPADDR(group address) PORT(port number)
```

donde *MC1* es el nombre del objeto COMMINFO, *dirección de grupo* es la dirección IP o el nombre DNS de multidifusión de grupo y *número de puertos* el puerto en el que se va a transmitir (el valor predeterminado es 1414).

Se crea un nuevo objeto COMMINFO denominado *MC1*; este nombre es el que debe especificar al definir un objeto TOPIC en el ejemplo siguiente.

#### Creación de un objeto **TOPIC** para multidifusión

Un tema es el asunto de la información que se publica en un mensaje de publicación/suscripción y un tema se define creando un objeto TOPIC. Los objetos TOPIC tienen dos parámetros que definen si pueden utilizarse con multidifusión o no. Estos parámetros son: **COMMINFO** y **MCAST**.

- **COMMINFO** Este parámetro especifica el nombre del objeto de información de comunicación de multidifusión. Para obtener más información sobre los parámetros de objeto COMMINFO, consulte [DEFINE COMMINFO](#).
- **MCAST** Este parámetro especifica si la multidifusión está permitida en esta posición del árbol de temas.

Utilice el siguiente ejemplo de línea de mandatos para definir un objeto TOPIC para multidifusión:

```
DEFINE TOPIC(ALLSPORTS) TOPICSTR('Sports') COMMINFO(MC1) MCAST(ENABLED)
```

Se crea un nuevo objeto TOPIC denominado *ALLSPORTS*. Tiene una serie de tema *Sports*, su objeto de información de comunicación relacionado se denomina *MC1* (que es el nombre que ha especificado al definir un objeto COMMINFO en el ejemplo anterior), y la multidifusión está habilitada.

#### Prueba de la publicación/suscripción de multidifusión

Después de que se hayan creado los objetos TOPIC y COMMINFO, se pueden probar utilizando el ejemplo *amqspubc* y el ejemplo *amqssubc*. Para obtener más información sobre estos ejemplos, consulte [Los programas de ejemplo de publicación/suscripción](#).

1. Abra dos ventanas de línea de mandatos; la primera línea de mandatos es para el ejemplo de publicación *amqspubc* y la segunda línea de mandatos es para el ejemplo de suscripción *amqssubc*.

2. Entre el siguiente mandato en la línea de mandatos 1:

```
amqspubc Sports QM1
```

donde *Sports* es la serie de tema del objeto TOPIC definido en un ejemplo anterior, y *QM1* es el nombre del gestor de colas.

3. Entre el siguiente mandato en la línea de mandatos 2:

```
amqssubc Sports QM1
```

donde *Sports* y *QM1* son los mismos que se utilizaron en el paso “2” en la página 350.

4. Especifique `Hello world` en la línea de mandatos 1. Si el puerto y la dirección IP especificados en el objeto COMMINFO están configurados correctamente; el ejemplo `amqssubc`, que está a la escucha en el puerto de las publicaciones de la dirección especificada, genera `Hello world` en la línea de mandatos 2.

## Topología de temas de IBM MQ Multicast

Utilice este ejemplo para entender la topología de temas de IBM MQ Multicast.

El soporte de IBM MQ Multicast requiere que cada subárbol tenga su propio grupo y corriente de datos de multidifusión dentro de la jerarquía total.

El esquema de direccionamiento IP *classful network* ha designado espacio de direcciones para la dirección de multidifusión. El rango completo de multidifusión de la dirección IP es 224.0.0.0 a 239.255.255.255, pero algunas de estas direcciones están reservadas. Para obtener una lista de direcciones reservadas, póngase en contacto con el administrador del sistema o consulte <https://www.iana.org/assignments/multicast-addresses> para obtener más información. Se recomienda utilizar la dirección de multidifusión con ámbito local en el rango de 239.0.0.0 a 239.255.255.255.

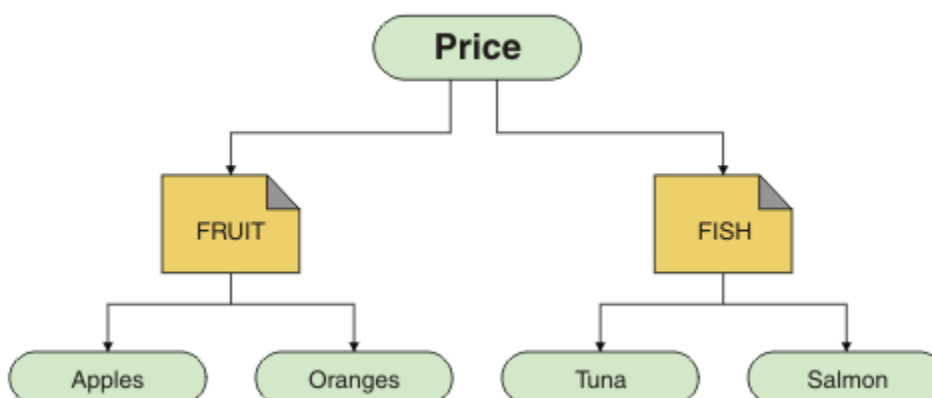
En el diagrama siguiente, hay dos posibles corrientes de datos de multidifusión:

```
DEF COMMINFO(MC1) GRPADDR(239.XXX.XXX.XXX
)
DEF COMMINFO(MC2) GRPADDR(239.YYY.YYY.YYY)
```

donde 239.XXX.XXX.XXX y 239.YYY.YYY.YYY son direcciones de multidifusión válidas.

Estas definiciones de temas se utilizan para crear un árbol de temas tal como se muestra en el diagrama siguiente:

```
DEFINE TOPIC(FRUIT) TOPICSTRING('Price/FRUIT') MCAST(ENABLED) COMMINFO(MC1)
DEFINE TOPIC(FISH) TOPICSTRING('Price/FISH') MCAST(ENABLED) COMMINFO(MC2)
```



Cada objeto de información de comunicación de multidifusión (COMMINFO) representa una corriente de datos diferente debido a que las direcciones de grupo son diferentes. En este ejemplo, el tema FRUIT se ha definido para utilizar el objeto COMMINFO MC1, el tema FISH se ha definido para utilizar el objeto COMMINFO MC2 y el nodo Price no tiene definiciones de multidifusión.

IBM MQ Multicast tiene un límite de 255 caracteres para las series de tema. Esta limitación significa que hay que tener cuidado con los nombres de nodos y los nodos hoja dentro del árbol; si los nombres de nodos y los nodos hoja son demasiado largos, la serie de tema puede superar los 255 caracteres y devolver el código de razón de 2425 (0979) (RC2425): MQRC\_TOPIC\_STRING\_ERROR. Es recomendable que las series de tema sean lo más cortas posible porque unas series de tema más largas pueden tener un efecto perjudicial en el rendimiento.

## Control del tamaño de mensajes de multidifusión

Utilice esta información para conocer el formato de mensaje de IBM MQ, y para reducir el tamaño de los mensajes de IBM MQ.

Los mensajes de IBM MQ tienen varios atributos asociados con ellos que están contenidos en el descriptor de mensaje. Para los mensajes pequeños, estos atributos pueden representar la mayor parte del tráfico de datos y pueden tener un importante efecto perjudicial sobre la velocidad de transmisión. IBM MQ Multicast permite al usuario configurar cuáles de estos atributos, si los hay, se transmiten junto con el mensaje.

La presencia de atributos de mensaje, distintos de la serie de tema, depende de si el objeto COMMINFO indica que deben enviarse o no. Si un atributo no se transmite, la aplicación receptora aplica un valor predeterminado. Los valores MQMD predeterminados no son necesariamente los mismos que el valor MQMD\_DEFAULT y se describen en [Tabla 24 en la página 352](#).

El objeto COMMINFO contiene el atributo MCPR0P, que controla cuántos campos MQMD y propiedades de usuario fluyen con el mensaje. Al establecer el valor de este atributo en un nivel apropiado, puede controlar el tamaño de los mensajes de IBM MQ Multicast:

### **MCPR0P**

Las propiedades multidifusión controla cuántas de las propiedades MQMD y de las propiedades de usuario fluyen con el mensaje.

#### **ALL**

Todas las propiedades de usuario y todos los campos de MQMD se transmiten.

#### **REPLY**

Sólo se transmiten las propiedades de usuario y los campos MQMD que están relacionados con la respuesta a los mensajes. Estas propiedades son:

- MsgType
- MessageId
- CorrelId
- ReplyToQ
- ReplyToQmgr

#### **USER**

Sólo se transmiten las propiedades de usuario

#### **NINGUNO**

No se transmiten las propiedades de usuario ni los campos MQMD

#### **COMPAT**

Este valor hace que la transmisión del mensaje se realice en una modalidad compatible para RMM, que permite determinada interoperación con las aplicaciones XMS actuales y las aplicaciones IBM Integration Bus RMM.

## Atributos de los mensajes de multidifusión

Los atributos de mensaje puede proceder de varios lugares, como MQMD, los campos de MQRFH2 y propiedades de mensaje.

La tabla siguiente muestra lo que sucede cuando los mensajes se envían sujetos al valor de MCPROP y el valor predeterminado utilizado cuando no se envía un atributo.

*Tabla 24. Atributos de mensajería y cómo se relacionan con la multidifusión*

<b>Atributo</b>	<b>Acción cuando se utiliza multidifusión</b>	<b>Valor predeterminado si no se transmite</b>
TopicString	Siempre incluido	No aplicable
MQMQ StrucId	No transmitido	No aplicable
MQMD Version	No transmitido	No aplicable
Informe	Incluido si no es predeterminado	0
MsgType	Incluido si no es predeterminado	MQMT_DATAGRAM
Caducidad	Incluido si no es predeterminado	0
Comentarios	Incluido si no es predeterminado	0
Codificación	Incluido si no es predeterminado	MQENC_NORMAL(equiv)
CodedCharSetId	Incluido si no es predeterminado	1208
Formato	Incluido si no es predeterminado	MQRFH2
Priority	Incluido si no es predeterminado	4
Persistence	Incluido si no es predeterminado	MQPER_NOT_PERSISTENT
MsgId	Incluido si no es predeterminado	Null
CorrelId	Incluido si no es predeterminado	Null
BackoutCount	Incluido si no es predeterminado	0
ReplyToQ	Incluido si no es predeterminado	Espacio en blanco
ReplyToQMgr	Incluido si no es predeterminado	Espacio en blanco
UserIdentifier	Incluido si no es predeterminado	Espacio en blanco
AccountingToken	Incluido si no es predeterminado	Null
PutAppIType	Incluido si no es predeterminado	MQAT_JAVA
PutAppIName	Incluido si no es predeterminado	Espacio en blanco
PutDate	Incluido si no es predeterminado	Espacio en blanco
PutTime	Incluido si no es predeterminado	Espacio en blanco
ApplOriginData	Incluido si no es predeterminado	Espacio en blanco
GroupID	Excluidos	No aplicable
MsgSeqNumber	Excluidos	No aplicable
Desplazamiento	Excluidos	No aplicable
MsgFlags	Excluidos	No aplicable
OriginalLength	Excluidos	No aplicable



Tabla 24. Atributos de mensajería y cómo se relacionan con la multidifusión (continuación)

Atributo	Acción cuando se utiliza multidifusión	Valor predeterminado si no se transmite
UserProperties	Incluidos	No aplicable

#### Referencia relacionada

[ALTER COMMINFO](#)

[DEFINE COMMINFO](#)

## Habilitación de la conversión de datos para la mensajería de Multicast

Utilice esta información para entender cómo funciona la conversión de datos para la mensajería de IBM MQ Multicast.

IBM MQ Multicast es un protocolo sin conexión compartido y, por tanto, no es posible que cada cliente realice solicitudes específicas para la conversión de datos. Todos los clientes suscritos a la misma secuencia de multidifusión reciben los mismos datos binarios; por lo tanto, si es necesaria la conversión de datos de IBM MQ, esta se realiza localmente en cada cliente.

En una instalación de plataforma mixta, es posible que la mayoría de los clientes requieran los datos en un formato que no sea el formato nativo de la aplicación transmisora. En esta situación, los valores de **CCSID** y **ENCODING** del objeto COMMINFO de multidifusión se pueden utilizar para definir la codificación de la transmisión de mensajes para conseguir eficiencia.

IBM MQ Multicast da soporte a la conversión de datos de la carga útil de mensaje para los siguientes formatos incorporados:

- MQADMIN
- MQEVENT
- MQPCF
- MQRFH
- MQRFH2
- MQSTR

Además de estos formatos, también puede definir sus propios formatos y utilizar una salida de conversión de datos de [MQDXP: Parámetro de salida de conversión de datos](#).

Si desea información sobre la programación de conversiones de datos, consulte [Conversión de datos en MQI para la mensajería de multidifusión](#).

Para obtener más información sobre la conversión de datos, consulte [Conversión de datos](#).

Si desea más información sobre las salidas de la conversión de datos y `ClientExitPath`, consulte [Stanza ClientExitPath del archivo de configuración de cliente](#).

## Supervisión de aplicaciones de multidifusión

Utilice esta información para aprender sobre la administración y la supervisión de IBM MQ Multicast.

El estado de los publicadores y suscriptores actuales para el tráfico de multidifusión (por ejemplo, el número de mensajes enviados y recibidos, o el número de mensajes perdidos) se transmite periódicamente al servidor desde el cliente. Cuando se recibe el estado, el atributo `COMMEV` del objeto COMMINFO especifica si el gestor de colas transmite o no un mensaje de suceso en `SYSTEM.ADMIN.PUBSUB.EVENT`. El mensaje de suceso contiene la información de estado recibida. Esta información es una inestimable ayuda de diagnóstico para buscar el origen de un problema.

Utilice el mandato `MQSC DISPLAY CONN` para visualizar información de conexión sobre las aplicaciones conectadas al gestor de colas. Para obtener más información sobre el mandato `DISPLAY CONN`, consulte [DISPLAY CONN](#).

Utilice el mandato MQSC **DISPLAY TPSTATUS** para visualizar el estado de los publicadores y suscriptores. Para obtener más información sobre el mandato **DISPLAY TPSTATUS**, consulte [DISPLAY TPSTATUS](#).

## COMMEV y el indicador de la fiabilidad de mensajes de multidifusión

El *indicador de confiabilidad*, utilizado junto con el atributo **COMMEV** del objeto COMMINFO, es un elemento clave en la supervisión de publicadores y suscriptores de IBM MQ Multicast. El indicador de fiabilidad (el campo **MSGREL** que se devuelve en los mandatos de estado de Publicación o Suscripción) es un indicador de IBM MQ que muestra el porcentaje de transmisiones que no tienen errores. A veces se tienen que retransmitir mensajes debido a un error de transmisión, lo que se refleja en el valor de **MSGREL**. Las causas potenciales de errores de transmisión incluyen los suscriptores lentos, redes ocupadas e interrupciones de la red. **COMMEV** controla si se generan mensajes de suceso para manejadores de multidifusión que se crean mediante el objeto COMMINFO y se establece en uno de tres valores posibles:

### DISABLED

Los mensajes de suceso no se graban.

### ENABLED

Los mensajes de suceso se graban siempre, con una frecuencia definida en el parámetro COMMINFO **MONINT**.

### EXCEPTION

Los mensajes de suceso se graban si la fiabilidad de los mensajes es inferior al umbral de fiabilidad. Un nivel de fiabilidad de mensaje de 90% o menos indica que puede existir un problema con la configuración de red, o que una o varias de las aplicaciones de publicación/suscripción se ejecuta demasiado lentamente:

- El valor **MSGREL (100, 100)** indica que no ha habido problemas en el período de tiempo de corto plazo o largo plazo.
- El valor **MSGREL (80, 60)** indica que el 20% de los mensajes tienen problemas actualmente, pero también una mejora del valor a largo plazo de 60.

Los clientes pueden continuar transmitiendo y recibiendo tráfico de multidifusión incluso cuando se interrumpe la conexión de difusión única para el gestor de colas, por lo tanto, los datos pueden estar desfasados.

## Fiabilidad de los mensajes de multidifusión

Utilice esta información para aprender a establecer el historial de suscripciones y de mensajes de IBM MQ Multicast.

Un elemento clave para superar una anomalía de transmisión con la multidifusión es el almacenamiento en búfer de los datos transmitidos (un historial de mensajes que se mantiene en el extremo transmisor del enlace) por parte de IBM MQ. Este proceso significa que no es necesario ningún almacenamiento en búfer de mensajes en el proceso de la aplicación transmisora, ya que IBM MQ proporciona la fiabilidad. El tamaño de este historial se configura mediante el objeto de información de comunicación (COMMINFO), tal como se describe en la siguiente información. Un almacenamiento intermedio de transmisión mayor significa que hay más historial de transmisión que retransmitir si es necesario, pero debido a la naturaleza de la multidifusión, no se puede dar soporte a una garantía del 100% de entrega.

El historial de mensajes de IBM MQ Multicast se controla en el objeto de información de comunicación (COMMINFO) mediante el atributo **MSGHIST**:

### MSGHIST

Este valor es la cantidad de historial de mensajes en kilobytes que mantiene el sistema para manejar las retransmisiones en el caso de acuses de recibo negativos (NACK).

El valor 0 ofrece el nivel mínimo de fiabilidad. El valor predeterminado es 100 KB.

El historial de nuevas suscripciones de IBM MQ Multicast se controla en el objeto de información de comunicación (COMMINFO) mediante el atributo **NSUBHIST**:

## NSUBHIST

El historial de nuevas suscripciones controla si un suscriptor que se suscribe a una corriente de datos de publicación recibe todos los datos disponibles en este momento, o bien recibe únicamente publicaciones desde el momento de la suscripción.

## NINGUNO

El valor NONE hace que el transmisor transmita sólo las publicaciones realizadas desde el momento de la suscripción. NONE es el valor predeterminado.

## ALL

Un valor ALL hace que el transmisor retransmita todo el historial del tema que sea conocido. En algunas circunstancias, esta situación puede proporcionar un comportamiento similar a las publicaciones retenidas.

**Nota:** La utilización del valor ALL puede tener un efecto perjudicial en el rendimiento si hay un historial de tema extenso, ya que se retransmite todo el historial del tema.

## Referencia relacionada

[DEFINE COMMINFO](#)

[ALTER COMMINFO](#)

## Tareas avanzadas de multidifusión

Utilice esta información para obtener información sobre las tareas avanzadas de administración de multidifusión de IBM MQ, como la configuración de archivos `.ini` y la interoperatividad con IBM MQ LLM.

Si desea ver consideraciones de seguridad en una instalación de multidifusión, consulte [Seguridad de multidifusión](#).

## Puente entre dominios de publicación/suscripción de multidifusión y no de multidifusión

Utilice esta información para entender lo que sucede cuando un publicador no de multidifusión publica en un tema habilitado para IBM MQ Multicast.

Si un publicador no de multidifusión publica en un tema definido como habilitado para **MCAST** y para **BRIDGE**, el gestor de colas transmite el mensaje a través de multidifusión directamente a los suscriptores que puedan estar escuchando. Un publicador de multidifusión no puede publicar en los temas que no están habilitados para la multidifusión.

Los temas existentes pueden estar habilitados para multidifusión si se establecen los parámetros **MCAST** y **COMMINFO** de un objeto de tema. Consulte [Conceptos de multidifusión iniciales](#) para obtener más información sobre estos parámetros.

El atributo **BRIDGE** del objeto COMMINFO controla las publicaciones de publicaciones que no utilizan multidifusión. Si **BRIDGE** se establece en ENABLED y el parámetro **MCAST** del tema también se establece en ENABLED, las publicaciones de las aplicaciones que no están utilizando multidifusión se usan como puente con las aplicaciones que sí. Para obtener más información sobre el parámetro **BRIDGE**, consulte [DEFINE COMMINFO](#).

## Configuración de los archivos .ini para Multicast

Utilice esta información para entender los campos de multidifusión de IBM MQ en los archivos `.ini`.

La configuración adicional de IBM MQ Multicast se puede realizar en un archivo `ini`. El archivo `ini` específico que debe utilizar depende del tipo de aplicaciones:

- Cliente: Configure el archivo `MQ_DATA_PATH/mqclient.ini`.
- Gestor de colas: Configure el archivo `MQ_DATA_PATH/qmgrs/QMNAME/qm.ini`.

donde `MQ_DATA_PATH` es la ubicación del directorio de datos IBM MQ (`/var/mqm/mqclient.ini`), y `QMNAME` es el nombre del gestor de colas al que se aplica el archivo `.ini`.

El archivo `.ini` contiene campos utilizados para ajustar el comportamiento de multidifusión de IBM MQ:

```
Multicast:
Protocol      = IP | UDP
IPVersion     = IPv4 | IPv6 | ANY | BOTH
LimitTransRate = DISABLED | STATIC | DYNAMIC
TransRateLimit = 100000
SocketTTL     = 1
Batch        = NO
Loop         = 1
Interface     = <IPAddress>
FeedbackMode  = ACK | NACK | WAIT1
HeartbeatTimeout = 20000
HeartbeatInterval = 2000
```

## Protocolo

### UDP

En esta modalidad, los paquetes se envían utilizando el protocolo UDP. Los elementos de la red no pueden proporcionar asistencia en la distribución de multidifusión como lo hacen en la modalidad IP. El formato de paquete sigue siendo compatible con PGM. Éste es el valor predeterminado.

### IP

En esta modalidad, el transmisor envía paquetes IP sin formato. Los elementos de red con soporte PGM ayudan en la distribución de paquetes de multidifusión fiables. Esta modalidad es totalmente compatible con el estándar PGM.

## IPVersion

### IPv4

Comunicarse utilizando sólo el protocolo IPv4. Éste es el valor predeterminado.

### IPv6

Comunicarse utilizando sólo el protocolo IPv6.

### CUALQUIERA

Comunicarse utilizando IPv4, IPv6, o ambos, dependiendo de qué protocolo está disponible.

### BOTH

Admite la comunicación utilizando IPv4 e IPv6.

## LimitTransRate

### DISABLED

No hay ningún control de velocidad de transmisión. Éste es el valor predeterminado.

### STATIC

Implementa el control de velocidad de transmisión estático. El transmisor no podría transmitir a una velocidad que supere la especificada por el parámetro `TransRateLimit`.

### DYNAMIC

El transmisor adapta su velocidad de transmisión de acuerdo con los comentarios que obtiene de los receptores. En este caso, el límite de la velocidad de transmisión no puede ser mayor que el valor especificado por el parámetro `TransRateLimit`. El transmisor intenta alcanzar una velocidad de transmisión óptima.

## TransRateLimit

Límite de la velocidad de transmisión en Kbps.

## SocketTTL

El valor de `SocketTTL` determina si el tráfico de multidifusión puede pasar a través de un direccionador, o el número de direccionadores por los que puede pasar a través.

## Lote

Controla si los mensajes se envían por lotes o inmediatamente; hay 2 valores posibles:

- *NO* Los mensajes no se envían por lotes, se envían inmediatamente.
- *YES* Los mensajes se envían por lotes.

## Loop

Establezca el valor en 1 para habilitar el bucle de multidifusión. El bucle de multidifusión define si los datos enviados recorren un bucle de retorno al host o no.

## Interface

Dirección IP de la interfaz en la que fluye el tráfico de multidifusión. Para obtener más información y solucionar problemas, consulte: [Prueba de aplicaciones de multidifusión en una red de no multidifusión](#) y [Establecimiento de la red adecuada para el tráfico de multidifusión](#)

## FeedbackMode

### NACK

Comentarios por acuses de recibo negativos. Éste es el valor predeterminado.

### ACK

Comentarios por acuses de recibo positivos.

### WAIT1

Comentarios por acuses de recibo positivos cuando el transmisor sólo espera 1 ACK de cualquiera de los receptores.

## HeartbeatTimeout

Tiempo de espera de pulsaciones en milisegundos. Un valor de 0 indica que los sucesos de tiempo de espera de pulsaciones no están planteadas por el receptor o los receptores del tema. El valor predeterminado es 20000.

## HeartbeatInterval

Intervalo de pulsaciones en milisegundos. Un valor de 0 indica que no se envían pulsaciones. El intervalo de pulsaciones debe ser considerablemente más pequeño que el valor **HeartbeatTimeout** para evitar sucesos de tiempo de espera de pulsación falsos. El valor predeterminado es 2000.

## Interoperatividad de multidifusión con IBM MQ Low Latency Messaging

Utilice esta información para entender la interoperatividad entre IBM MQ Multicast e IBM MQ Low Latency Messaging (LLM).

La transferencia de carga útil básica es posible para una aplicación que utilice LLM, con otra aplicación que utilice la multidifusión para intercambiar mensajes en ambas direcciones. Aunque la multidifusión utiliza la tecnología LLM, el producto LLM en sí no está incorporado. Por lo tanto, es posible instalar LLM e IBM MQ Multicast, y operar y dar servicio a los dos productos por separado.

Las aplicaciones LLM que se comunican con la multidifusión quizá tengan que enviar y recibir propiedades de mensaje. Las propiedades de mensaje de IBM MQ y los campos MQMD se transmiten como propiedades de mensaje LLM con códigos de propiedad de mensaje LLM específicos tal como se muestra en la siguiente tabla:

Propiedad de IBM MQ	Tipo de propiedad de IBM MQ LLM	Clase de propiedad de LLM	Código de propiedad de LLM
MQMD.Report	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1001
MQMD.MsgType	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1002
MQMD.Expiry	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1003
MQMD.Feedback	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1004
MQMD.Encoding	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1005
MQMD.CodedCharSetId	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1006
MQMD.Format	RMM_MSG_PROP_BYTES	LLM_PROP_KIND_String	-1007
MQMD.Priority	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1008

Tabla 25. Correlaciones de propiedades de mensaje de IBM MQ con propiedades de IBM MQ LLM (continuación)

Propiedad de IBM MQ	Tipo de propiedad de IBM MQ LLM	Clase de propiedad de LLM	Código de propiedad de LLM
MQMD.Persistence	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1009
MQMD.MsgId	RMM_MSG_PROP_BYTES	LLM_PROP_KIND_ByteArray	-1010
MQMD.BackoutCount	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1012
MQMD.ReplyToQ	RMM_MSG_PROP_BYTES	LLM_PROP_KIND_String	-1013
MQMD.ReplyToQMger	RMM_MSG_PROP_BYTES	LLM_PROP_KIND_String	-1014
MQMD.PutDate	RMM_MSG_PROP_BYTES	LLM_PROP_KIND_String	-1020
MQMD.PutTime	RMM_MSG_PROP_BYTES	LLM_PROP_KIND_String	-1021
MQMD.ApplOriginData	RMM_MSG_PROP_BYTES	LLM_PROP_KIND_String	-1022
MQPubOptions	RMM_MSG_PROP_INT32	LLM_PROP_KIND_int32	-1053

Para obtener más información sobre LLM, consulte la documentación del producto LLM: [Mensajería de baja latencia de IBM MQ](#).

## IBM i Administración de IBM MQ for IBM i

Presenta los métodos de los que dispone para administrar IBM MQ en IBM i.

Las tareas de administración incluyen la creación, inicio, modificación, visualización, detención y supresión de clústeres, procesos y objetos de IBM MQ (gestores de colas, colas, listas de nombres, definiciones de proceso, canales, canales de conexión de clientes, escuchas, servicios y objetos de información de autenticación).

Consulte los siguientes enlaces para obtener información detallada sobre cómo administrar IBM MQ for IBM i:

- [“Gestión de IBM MQ for IBM i utilizando mandatos CL” en la página 359](#)
- [“Formas alternativas de administrar IBM MQ for IBM i” en la página 372](#)
- [“Gestión de trabajo en IBM i” en la página 378](#)

### Conceptos relacionados

[“Disponibilidad, copia de seguridad, recuperación y reinicio en IBM i” en la página 385](#)

Utilice esta información para entender cómo IBM MQ for IBM i utiliza el soporte de registro por diario de IBM i como ayuda para su estrategia de copia de seguridad y restauración.

[Explicación de los nombres de biblioteca de gestor de colas de IBM MQ for IBM i](#)

[Servicios instalables y componentes en IBM i](#)

### Tareas relacionadas

[Modificación de la información de configuración en IBM i](#)

[Configuración de la seguridad en IBM i](#)

[Determinación de problemas con aplicaciones IBM MQ for IBM i](#)

### Referencia relacionada

[“Progresivo IBM MQ for IBM i” en la página 430](#)

En esta sección se explica cómo desactivar temporalmente (finalizar ordenadamente) IBM MQ for IBM i.

[“El manejador de la cola de mensajes no entregados en IBM i” en la página 169](#)

Qué es el manejador de la cola de mensajes no entregados de IBM i y cómo invocarlo

[Objetos predeterminados y del sistema](#)

## Gestión de IBM MQ for IBM i utilizando mandatos CL

Utilice esta información para obtener una descripción de los mandatos de IBM MQ IBM i.

Se puede acceder a la mayoría de los grupos de mandatos de IBM MQ, incluidos los que están asociados a gestores de colas, colas, temas, canales, listas de nombres, definiciones de proceso y objetos de información de autenticación, mediante el mandato **WRK\*** correspondiente.

El principal mandato del grupo es **WRKMQM**. Este mandato le permite, por ejemplo, visualizar una lista de todos los gestores de colas del sistema, junto con la información de estado. De forma alternativa, puede procesar todos los mandatos específicos del gestor de colas utilizando distintas opciones con respecto a cada entrada.

Desde el mandato **WRKMQM** puede seleccionar áreas específicas de cada gestor de colas, por ejemplo, trabajar con canales, temas o colas, y a partir de ahí seleccionar objetos individuales.

### Registrar definiciones de aplicaciones de IBM MQ

Al crear o personalizar aplicaciones de IBM MQ, resulta útil mantener un registro de todas las definiciones de IBM MQ que se han creado. Este registro se puede utilizar:

- Fines de recuperación
- Mantenimiento
- Retrotracción de las aplicaciones de IBM MQ

Puede registrar las definiciones de aplicaciones de IBM MQ de una de estas dos maneras:

1. Creando programas CL para generar las definiciones de IBM MQ para el servidor.
2. Creando archivos de texto MQSC como miembros SRC para generar las definiciones de IBM MQ utilizando el lenguaje de mandatos de IBM MQ para múltiples plataformas.

Para obtener más detalles sobre la definición de objetos de cola, consulte [“Administración utilizando mandatos MQSC”](#) en la página 12 y [“Utilización de los formatos de mandato programable de IBM MQ”](#) en la página 26.

#### Referencia relacionada

[Referencia de mandatos CL de IBM MQ for IBM i](#)

## Antes de empezar a utilizar IBM MQ for IBM i utilizando mandatos CL

Utilice esta información para iniciar el subsistema de IBM MQ y crear un gestor de colas local.

### Antes de empezar

Asegúrese de que el subsistema de IBM MQ está en ejecución (mediante el mandato STRSBS QMQM/QMQM) y que la cola de trabajos asociada a ese subsistema no está retenida. De forma predeterminada, el subsistema de IBM MQ y la cola de trabajos se llaman ambos QMQM en la biblioteca QMQM.

### Acerca de esta tarea

Utilizar la línea de mandatos de IBM i para iniciar un gestor de colas

### Procedimiento

1. Cree un gestor de colas local emitiendo el mandato CRTMQM desde una línea de mandatos de IBM i.  
Al crear un gestor de colas, tiene la opción de hacer que ese gestor de colas sea el gestor de colas predeterminado. El gestor de colas predeterminado (sólo puede haber uno) es aquel al que se aplica un mandato CL, si se omite el parámetro del nombre del gestor de colas (MQMNAME).
2. Inicie un gestor de colas local emitiendo el mandato STRMQM desde una línea de mandatos de IBM i.

Si el gestor de colas tarda más de algunos segundos en reiniciarse, IBM MQ mostrará mensajes de estado de forma intermitente que detallan el progreso del inicio. Para obtener más información sobre estos mensajes, consulte [Mensajes y códigos de razón](#).

## Qué hacer a continuación

Puede detener un gestor de colas emitiendo el mandato ENDMQM desde la línea de mandatos de IBM i, y controlar un gestor de colas emitiendo otros mandatos de IBM MQ desde una línea de mandatos de IBM i.

Los gestores de colas remotos no se pueden iniciar de forma remota, sino que han de ser los operadores locales quienes los creen y los inicien en sus sistemas. Una excepción a esta regla es cuando existen recursos de operación remota (externos a IBM MQ for IBM i) que permiten habilitar estas operaciones.

El administrador de colas local no puede detener un gestor de colas remoto.

**Nota:** Como parte de la desactivación temporal de un sistema IBM MQ, tiene que desactivar temporalmente los gestores de colas activos. Este tema se describe en el apartado [“Progresivo IBM MQ for IBM i”](#) en la página 430.

## IBM i Crear objetos de IBM MQ for IBM i

Utilice esta información para comprender los métodos para crear objetos de IBM MQ para IBM i.

### Antes de empezar

Las siguientes tareas sugieren diversas maneras de utilizar IBM MQ for IBM i desde la línea de mandatos.

### Acerca de esta tarea

Hay dos métodos en línea para crear objetos de IBM MQ, que son:

### Procedimiento

1. Utilizando un mandato Create, por ejemplo: el mandato **Create MQM Queue : CRTMQMQ**
2. Utilizando un mandato Trabajar con objeto MQM, seguido de F6, por ejemplo: El mandato **Work with MQM Queues : WRKMQMQ**

## Qué hacer a continuación

Para obtener una lista de todos los mandatos, consulte [Mandatos CL de IBM MQ for IBM i](#).

**Nota:** Todos los mandatos de MQM se pueden emitir desde el menú Mandatos de gestor de colas de mensajes. Para visualizar este menú, escriba GO CMDMQM en la línea de mandatos y pulse la tecla Enter .

El sistema muestra automáticamente el panel de solicitud cuando se selecciona un mandato de este menú. Para visualizar el panel de solicitud para un mandato que ha escrito directamente en la línea de mandatos, pulse F4 antes de pulsar la tecla Enter .

### Crear una cola local mediante el mandato CRTMQMQ

### Procedimiento

1. Escriba CHGMQM en la línea de mandatos y pulse la tecla F4.
2. En el panel **Crear cola MQM**, escriba el nombre de la cola que desea crear en el campo Queue name . Si desea especificar un nombre con una combinación de mayúsculas/minúsculas, encierre el nombre entre comillas simples.
3. Escriba \*LCL en el campo Queue type .
4. Especifique un nombre de gestor de colas, a menos que esté utilizando el gestor de colas predeterminado, y pulse la tecla Enter . Puede sobrescribir cualquiera de los valores con un nuevo



valor. Desplácese hacia delante para ver más campos. Las opciones usadas para los clústeres están al final de la lista de opciones.

5. Cuando haya cambiado algún valor, pulse la tecla `Enter` para crear la cola.

## Crear una cola local mediante el mandato **WRKMQM**

### Procedimiento

1. Escriba **WRKMQM** en la línea de mandatos.
2. Entre el nombre de un gestor de colas.
3. Si desea visualizar el panel de solicitud, pulse `F4`. El panel de solicitud es útil para reducir el número de colas visualizadas, especificando para ello un nombre de cola o un tipo de cola genérico.
4. Pulse `Enter` y se mostrará el **panel Trabajar con colas MQM**. Puede sobrescribir cualquiera de los valores con un nuevo valor. Desplácese hacia delante para ver más campos. Las opciones usadas para los clústeres están al final de la lista de opciones.
5. Pulse `F6` para crear una cola nueva; aparecerá el panel **CRTMQM**. Consulte [“Crear una cola local mediante el mandato CRTMQM”](#) en la página 360 para obtener instrucciones sobre cómo crear la cola. Cuando haya creado la cola, vuelve a aparecer el panel **Trabajar con colas MQM**. La nueva cola se añade a la lista cuando pulsa `F5=Refresh`.

### Modificación de los atributos del gestor de colas

#### Acerca de esta tarea

Para modificar los atributos del gestor de colas especificado en el mandato **CHGMQM**, indique los atributos y los valores que desea cambiar. Por ejemplo, para modificar los atributos de `jupiter.queue.manager`, se utilizan estas opciones:

### Procedimiento

Escriba **CHGMQM** en la línea de mandatos y pulse la tecla `F4`.

### Resultados

El mandato cambia la cola de mensajes no entregados utilizada y habilita la función de inhibir sucesos.

## Utilización de colas locales en IBM i

Esta sección contiene ejemplos de algunos de los mandatos que se pueden usar para gestionar colas locales. Todos los mandatos que se muestran también están disponibles utilizando las opciones del **panel de mandatos WRKMQM**.

### Definición de una cola local

Para una aplicación, el gestor de colas local es el gestor de colas al que está conectada la aplicación. Las colas gestionadas por el gestor de colas local son locales con respecto a ese gestor de colas.

Utilice el mandato **CRTMQM QTYPE \*LCL** para crear una definición de una cola local y también para crear la estructura de datos llamada cola. Además, puede modificar las características de la cola para que sean distintas de las de la cola local predeterminada.

En este ejemplo, la cola que definimos, `orange.local.queue`, se ha especificado para que tenga estas características:

- Está habilitada para las operaciones de obtención, inhabilitada para las operaciones de transferencia y funciona de acuerdo a primero en entrar, primero en salir (FIFO).
- Es una cola *corriente*, es decir, no es una cola de inicio ni una cola de transmisión, ni tampoco genera mensajes desencadenantes.

- La profundidad máxima de la cola es de 1000 mensajes; la longitud máxima de mensaje es de 2000 bytes.

Esto se logra con el siguiente mandato en el gestor de colas predeterminado:

```
CRTMQMQ QNAME('orange.local.queue') QTYPE(*LCL)
TEXT('Queue for messages from other systems')
PUTENBL(*NO)
GETENBL(*YES)
TRGENBL(*NO)
MSGDLYSEQ(*FIFO)
MAXDEPTH(1000)
MAXMSGLN(2000)
USAGE(*NORMAL)
```

#### Nota:

1. USAGE \*NORMAL indica que esta cola no es una cola de transmisión.
2. Si ya hay una cola local con el nombre `orange.local.queue` en el mismo gestor de colas, este mandato fallará. Utilice el atributo REPLACE \*YES si desea escribir encima de la definición ya existente de una cola, pero consulte también el apartado [“Modificación de atributos de cola local”](#) en la página 363.

## Definición de una cola de mensajes no entregados

Cada gestor de colas debe tener una cola local que se utilizará como cola de mensajes no entregados, con el fin de que los mensajes que no se puedan entregar en su destino correcto se almacenen para recuperación posterior. La cola de mensajes no entregados se debe indicar explícitamente en el gestor de colas. Para ello, especifique una cola de mensajes no entregados en el mandato **CRTMQM** o bien utilice el mandato **CHGMQM** para especificar una más adelante. Para poder utilizar la cola de mensajes no entregados, primero es preciso definirla.

Junto con el producto se suministra una cola de mensajes no entregados de ejemplo denominada SYSTEM.DEAD.LETTER.QUEUE. Esta cola se crea automáticamente cuando se crea el gestor de colas. Si es necesario, puede modificar esta definición. No es preciso cambiar su nombre, aunque puede hacerlo si lo desea.

Una cola de mensajes no entregados no tiene ningún requisito especial excepto que:

- Debe ser una cola local.
- Su atributo MAXMSGL (longitud máxima de mensajes) debe permitir que la cola pueda alojar los mensajes más grandes que el gestor de colas tenga que manejar **más** el tamaño de la cabecera de mensaje no entregado (MQDLH).

IBM MQ proporciona un manejador de la cola de mensajes no entregados que le permite especificar cómo se han de procesar o eliminar los mensajes encontrados en una cola de mensajes no entregados. Para obtener más información, consulte [“El manejador de la cola de mensajes no entregados en IBM i”](#) en la página 169.

## Visualización de los atributos de objeto predeterminados

Cuando define un objeto de IBM MQ, éste toma del objeto predeterminado todos los atributos que no especifique. Por ejemplo, cuando define una cola local, la cola hereda todos los atributos que omite en la definición de la cola local predeterminada, que tiene el nombre SYSTEM.DEFAULT.LOCAL.QUEUE. Si desea saber cuáles son exactamente esos atributos, utilice este mandato:

```
DSPMQMQ QNAME(SYSTEM.DEFAULT.LOCAL.QUEUE) MQMNAME(MYQUEUEMANAGER)
```

## Copiar una definición de cola local

Cuando desee copiar una definición de cola local, utilice el mandato CPYMQMQ. Por ejemplo:

```
CPYMQMQ FROMQ('orange.local.queue') TOQ('magenta.queue') MQMNAME(MYQUEUEMANAGER)
```

Este mandato crea una cola que tiene los mismos atributos que nuestra cola `orange.local.queue`, en vez de tener los atributos de la cola local predeterminada del sistema.

También puede utilizar el mandato **CPYMQMQ** para copiar una definición de cola, sustituyendo uno o varios cambios realizados en los atributos de la original. Por ejemplo:

```
CPYMQMQ FROMQ('orange.local.queue') TOQ('third.queue') MQMNAME(MYQUEUEMANAGER)  
MAXMSGLEN(1024)
```

Este mandato copia los atributos de la cola `orange.local.queue` en la cola `third.queue`, pero especifica que la longitud máxima de mensaje de la nueva cola ha de ser 1024 bytes, en lugar de 2000.

**Nota:** Cuando se utiliza el mandato **CPYMQMQ**, sólo se copian los atributos de la cola, no los mensajes de la cola.

## Modificación de atributos de cola local

Los atributos de las colas se pueden cambiar de dos formas: mediante el mandato **CHGMQMQ** o bien el mandato **CPYMQMQ** con el atributo `REPLACE *YES`. En [“Definición de una cola local”](#) en la página 361, ha definido la cola `orange.local.queue`. Si, por ejemplo, tiene que aumentar la longitud máxima de mensajes en esta cola a 10.000 bytes.

- Utilizando el mandato **CHGMQMQ**:

```
CHGMQMQ QNAME('orange.local.queue') MQMNAME(MYQUEUEMANAGER) MAXMSGLEN(10000)
```

Este mandato modifica un solo atributo, el de la longitud máxima del mensaje; todos los demás atributos no cambian.

- Si se utiliza el mandato **CRTMQMQ** con la opción `REPLACE *YES`, por ejemplo:

```
CRTMQMQ QNAME('orange.local.queue') QTYPE(*LCL) MQMNAME(MYQUEUEMANAGER)  
MAXMSGLEN(10000) REPLACE(*YES)
```

Este mandato no sólo cambia la longitud máxima de mensaje sino también los demás atributos, a los que se asignan sus valores predeterminados. La cola está ahora habilitada para operaciones de transferencia mientras que anteriormente estaba inhibida para dichas operaciones. El valor predeterminado es la transferencia habilita, según lo especificado por la cola `SYSTEM.DEFAULT.LOCAL.QUEUE`, a menos que se haya cambiado ese valor.

Si **reduce** la longitud máxima del mensaje en una cola existente, los mensajes existentes no se ven afectados. Sin embargo, todos los mensajes nuevos deben cumplir los nuevos criterios.

## Vaciar una cola local

Para suprimir todos los mensajes de una cola local denominada `magenta.queue`, utilice el mandato siguiente:

```
CLRMQMQ QNAME('magenta.queue') MQMNAME(MYQUEUEMANAGER)
```

Una cola no se puede vaciar si:

- Hay mensajes no confirmados que se han transferido a la cola bajo punto de sincronización.
- Hay una aplicación que tiene abierta actualmente la cola.

## Suprimir una cola local

Utilice el mandato **DLTMQM** para suprimir una cola local.

No se puede suprimir una cola si en ella hay mensajes no confirmados, o si se está utilizando.

## Habilitar colas grandes

IBM MQ da soporte a colas de más de 2 GB. Consulte la documentación del sistema operativo para obtener información sobre cómo permitir que IBM i dé soporte a archivos grandes.

La información del producto IBM i se puede encontrar en [IBM Documentation](#).

Es posible que algunos programas de utilidad no puedan manejar archivos de más de 2 GB. Antes de habilitar el soporte para archivos grandes, consulte la documentación de su sistema operativo para obtener información sobre las limitaciones de este soporte.

## IBM i Utilización de colas de alias en IBM i

Esta sección contiene ejemplos de algunos de los mandatos que se pueden usar para gestionar colas alias. Todos los mandatos que se muestran también están disponibles utilizando las opciones del **panel de mandatos WRKMQM**.

Una cola alias (a veces conocida como alias de cola) proporciona un método para redirigir las llamadas MQI. La cola alias no es una cola real, sino una definición que se resuelve en una cola real. La definición de cola alias contiene el nombre de una cola destino, que se especifica mediante el atributo TGTQNAME.

Cuando una aplicación especifica una cola alias en una llamada MQI, el gestor de colas resuelve el nombre de cola real en el momento de la ejecución.

Por ejemplo, supongamos que se ha desarrollado una aplicación para transferir los mensajes a una cola llamada `my.alias.queue`. La aplicación especifica el nombre de esta cola cuando realiza una solicitud **MQOPEN** e, indirectamente, si transfiere un mensaje a esta cola. La aplicación no sabe que la cola es una cola alias. Para cada llamada MQI que utilice este alias, el gestor de colas determina el nombre de cola real, que podría ser una cola local o una cola remota definida en este gestor de colas.

Si cambia el valor del atributo TGTQNAME, puede redirigir las llamadas MQI a otra cola, posiblemente situada en otro gestor de colas. Esto resulta de gran utilidad para el mantenimiento, la migración y el equilibrio de la carga.

## Definición de una cola alias

El mandato siguiente crea una cola alias:

```
CRTMQMQ QNAME('my.alias.queue') QTYPE(*ALS) TGTQNAME('yellow.queue')
MQMNAME(MYQUEUEMANAGER)
```

Este mandato redirige las llamadas MQI que especifican `my.alias.queue` a la cola `yellow.queue`. El mandato no crea la cola de destino; las llamadas MQI fallan si la cola `yellow.queue` no existe en el momento de la ejecución.

Si cambia la definición del alias, puede redirigir las llamadas MQI a otra cola. Por ejemplo:

```
CHGMQM QNAME('my.alias.queue') TGTQNAME('magenta.queue') MQMNAME(MYQUEUEMANAGER)
```

Este mandato redirige las llamadas MQI a otra cola, `magenta.queue`.

También puede utilizar colas alias para que hacer que una sola cola (la cola destino) parezca tener atributos distintos para aplicaciones distintas. Esto se hace definiendo dos alias, uno para cada aplicación. Suponga que tiene dos aplicaciones:

- La aplicación ALPHA puede transferir mensajes a `yellow.queue`, pero no puede obtener mensajes de ella.

- La aplicación BETA puede obtener mensajes de yellow.queue, pero no tiene autorización para transferir mensajes a ella.

Para lograrlo, utilice estos mandatos:

```
/* This alias is put enabled and get disabled for application ALPHA */
CRTMQMQ QNAME('alphas.alias.queue') QTYPE(*ALS) TGTQNAME('yellow.queue')
PUTENBL(*YES) GETENBL(*NO) MQMNAME(MYQUEUEMANAGER)

/* This alias is put disabled and get enabled for application BETA */
CRTMQMQ QNAME('betas.alias.queue') QTYPE(*ALS) TGTQNAME('yellow.queue')
PUTENBL(*NO) GETENBL(*YES) MQMNAME(MYQUEUEMANAGER)
```

ALPHA utiliza el nombre de la cola alphas.alias.queue en las llamadas MQI ; BETA utiliza el nombre de la cola betas.alias.queue. Ambas aplicaciones acceden a la misma cola, pero de forma diferente.

Cuando define las colas alias, puede utilizar el atributo REPLACE \*YES, de la misma manera que cuando lo utiliza con las colas locales.

## Utilización de otros mandatos con colas alias

Puede utilizar los correspondientes mandatos para visualizar o cambiar los atributos de colas alias. Por ejemplo:

```
* Display the alias queue's attributes */
DSPMQMQ QNAME('alphas.alias.queue') MQMNAME(MYQUEUEMANAGER)

/* ALTER the base queue name, to which the alias resolves. */
/* FORCE = Force the change even if the queue is open. */

CHMQMQ QNAME('alphas.alias.queue') TGTQNAME('orange.local.queue') FORCE(*YES)
MQMNAME(MYQUEUEMANAGER)
```



## Trabajar con colas de modelo en IBM i

Esta sección contiene ejemplos de algunos de los mandatos que se pueden usar para gestionar colas modelo. Todos los mandatos que se muestran también están disponibles utilizando las opciones del **panel de mandatos WRKMQM**.

Un gestor de colas crea una cola dinámica si recibe una llamada MQI de una aplicación que especifica un nombre de cola que se ha definido como cola modelo. El nombre de la nueva cola dinámica lo genera el gestor de colas cuando se crea la cola. Una cola modelo es una plantilla que especifica los atributos de las colas dinámicas que se creen a partir de ella.

Las colas modelo proporcionan a las aplicaciones un método práctico de crear colas cuando se necesitan.

### Definición de una cola modelo

Las colas modelo se definen con un conjunto de atributos del mismo modo en que se define una cola local. Las colas modelo y las colas locales tienen el mismo conjunto de atributos, excepto que en las colas modelo se puede especificar si las colas dinámicas creadas son temporales o persistentes. (Las colas permanentes se conservan tras reiniciar el gestor de colas, las colas temporales no). Por ejemplo:

```
CRTMQMQ QNAME('green.model.queue') QTYPE(*MDL) DFNTYPE(*PERMDYN)
```

Este mandato crea una definición de cola modelo. Desde el atributo DFNTYPE, las colas reales creadas a partir de esta plantilla son colas dinámicas permanentes. Los atributos no especificados se copian automáticamente de la cola predeterminada SYSTEM.DEFAULT.MODEL.QUEUE.

Cuando defina las colas modelo, puede utilizar el atributo REPLACE \*YES, de la misma manera que cuando lo utiliza con las colas locales.

## Utilización de otros mandatos con colas modelo

Puede utilizar los correspondientes mandatos para visualizar o alterar los atributos de una cola modelo. Por ejemplo:

```
/* Display the model queue's attributes */
DSPMQMQ MQMNAME(MYQUEUEMANAGER) QNAME('green.model.queue')

/* ALTER the model queue to enable puts on any */
/* dynamic queue created from this model. */
CHGMQM MQMNAME(MYQUEUEMANAGER) QNAME('blue.model.queue') PUTENBL(*YES)
```

## IBM i Utilización del desencadenamiento en IBM i

Utilice esta información para aprender sobre el desencadenamiento y las definiciones de proceso.

IBM MQ proporciona un recurso para iniciar una aplicación automáticamente cuando se cumplen ciertas condiciones en una cola. Un ejemplo de estas condiciones es cuando el número de mensajes en una cola alcanza un número especificado. Este recurso se denomina *desencadenamiento* y se describe detalladamente en [Desencadenamiento de canales](#).

### ¿Qué son los desencadenantes?

El gestor de colas define ciertas condiciones como sucesos desencadenantes constituyentes. Si se habilita el desencadenamiento para una cola y se produce un suceso desencadenante, el gestor de colas envía un mensaje desencadenante a una cola llamada cola de inicio. La presencia del mensaje desencadenante en la cola de inicio indica que se ha producido un suceso desencadenante.

Los mensajes desencadenantes generados por el gestor de colas no son persistentes. Como consecuencia, se reducen las anotaciones cronológicas (lo que mejora el rendimiento), y se minimizan los duplicados durante el reinicio, de manera que se mejora el tiempo de reinicio.

### ¿Qué es el supervisor de desencadenantes?

El programa que procesa la cola de inicio se denomina aplicación supervisora desencadenante y su función consiste en leer el mensaje desencadenante y realizar las acciones adecuadas, basándose en la información contenida en el mensaje desencadenante. Normalmente esta acción consistiría en iniciar alguna otra aplicación para procesar la cola que ha causado la generación del mensaje desencadenante. Desde el punto de vista del gestor de colas, la aplicación supervisora desencadenante no tiene ninguna particularidad especial, se trata de otra aplicación que lee mensajes de una cola (la cola de inicio).

## Modificación de los atributos de sometimiento de trabajos del supervisor de desencadenantes

El supervisor desencadenante que se suministra como mandato **STRMQMTRM** somete un trabajo para cada mensaje desencadenante utilizando la descripción de trabajo predeterminada del sistema, QDFTJOB. Esto tiene algunas limitaciones, ya que los trabajos sometidos siempre se llaman QDFTJOB y tienen los atributos de la descripción de trabajo predeterminada, incluida la lista de bibliotecas \*SYSVAL. IBM MQ proporciona un método para alterar temporalmente estos atributos. Por ejemplo, es posible personalizar los trabajos sometidos para que tengan nombres más significativos, tal como se indica a continuación:

1. En la descripción del trabajo, especifique la descripción que desee, por ejemplo los valores de las anotaciones cronológicas.
2. Especifique los Datos de entorno de la definición de proceso utilizada en el proceso desencadenante:

```
CHGMQMPRC PRCNAME(MY_PROCESS) MQMNAME(MHA3) ENVDATA ('JOB(MYLIB/TRIGJOB)')
```

El Supervisor desencadenante realiza un SBMJOB utilizando la descripción especificada.

Es posible alterar temporalmente otros atributos del SBMJOB especificando la palabra clave y el valor adecuados en los Datos de entorno de la definición de proceso. La única excepción a esta regla es la palabra clave CMD, porque en este caso el supervisor de desencadenantes completa el atributo. A continuación se muestra un ejemplo del mandato para especificar los Datos de entorno de la definición de proceso, en el que se van a modificar el nombre de trabajo y la descripción:

```
CHGMQMPRC PRCNAME(MY_PROCESS) MQMNAME(MHA3) ENVDATA ('JOB(MYLIB/TRIGJOB)')  
JOB(TRIGGER)')
```

## Definición de una cola de aplicación para el desencadenamiento

Una cola de aplicación es una cola local que las aplicaciones utilizan para el envío de mensajes, mediante la MQI. El desencadenamiento requiere definir varios atributos de cola en la cola de aplicación. El propio desencadenamiento está habilitado mediante el atributo TRGENBL.

En este ejemplo, se ha de generar un suceso desencadenante cuando haya 100 mensajes de prioridad 5 o superior en la cola local `motor.insurance.queue`, tal como se indica a continuación:

```
CRTMQMQ MQMNAME(MYQUEUEMANAGER) QNAME('motor.insurance.queue') QTYPE(*LCL)  
PRCNAME('motor.insurance.quote.process') MAXMSGLEN(2000)  
DFTMSGPST(*YES) INITQNAME('motor.ins.init.queue')  
TRGENBL(*YES) TRGTYPE(*DEPTH) TRGDEPTH(100) TRGMSGPTY(5)
```

donde los parámetros son:

### **MQMNAME(MYQUEUEMANAGER)**

Nombre del gestor de colas.

### **QNAME('motor.insurance.queue')**

El nombre de la cola de aplicación que se está definiendo.

### **PRCNAME('motor.insurance.quote.process')**

El nombre de la aplicación que el programa supervisor desencadenante debe iniciar.

### **MAXMSGLEN(2000)**

La longitud máxima de los mensajes de la cola.

### **DFTMSGPST(\*YES)**

Los mensajes de esta cola son persistentes de forma predeterminada.

### **INITQNAME('motor.ins.init.queue')**

El nombre de la cola de inicio en la que el gestor de colas va a poner el mensaje desencadenante.

### **TRGENBL(\*YES)**

El valor del atributo de desencadenamiento.

### **TRGTYPE(\*DEPTH)**

Se genera un suceso desencadenante cuando el número de mensajes de la prioridad requerida (**TRGMSGPTY**) alcanza el número especificado en **TRGDEPTH**.

### **TRGDEPTH(100)**

El número de mensajes necesarios para generar un suceso desencadenante.

### **TRGMSGPTY(5)**

La prioridad de los mensajes que el gestor de colas va a contar para decidir si debe generar un suceso desencadenante. Sólo se incluyen en el recuento los mensajes con prioridad 5 o superior.

## Definición de una cola de inicio

Cuando se produce un suceso desencadenante, el gestor de colas coloca un mensaje desencadenante en la cola de inicio especificada en la definición de la cola de aplicación. Las colas de inicio no tienen ningún valor especial, pero la siguiente definición de la cola local `motor.ins.init.queue` le puede servir de orientación:

```
CRTMQMQ MQMNAME(MYQUEUEMANAGER) QNAME('motor.ins.init.queue') QTYPE(*LCL)
GETENBL(*YES) SHARE(*NO) TRGTYPE(*NONE)
MAXMSGL(2000)
MAXDEPTH(1000)
```

## Crear una definición de proceso

Para crear una definición de proceso, utilice el mandato **CRTMQMPCRC**. Una definición de proceso asocia una cola de aplicación a la aplicación que va a procesar los mensajes de la cola. Esto se lleva a cabo mediante el atributo `PRCNAME` en la cola de la aplicación `motor.insurance.queue`. El mandato siguiente crea el proceso necesario, `motor.insurance.quote.process`, identificado en este ejemplo:

```
CRTMQMPCRC MQMNAME(MYQUEUEMANAGER) PRCNAME('motor.insurance.quote.process')
TEXT('Insurance request message processing')
APPTYPE(*OS400) APPID(MQTEST/TESTPROG)
USRDATA('open, close, 235')
```

donde los parámetros son:

### **MQMNAME(MYQUEUEMANAGER)**

Nombre del gestor de colas.

### **PRCNAME('motor.insurance.quote.process')**

El nombre de la definición de proceso.

### **TEXT('Insurance request message processing')**

Una descripción del programa de aplicación relacionado con esta definición. Este texto se visualiza al utilizar el mandato **DSPMQMPCRC**. sirve para ayudarle a identificar lo que hace el proceso. Si utiliza espacios en la serie, debe colocarla entre comillas simples.

### **APPTYPE(\*OS400)**

El tipo de aplicación que se ha de iniciar.

### **APPID(MQTEST/TESTPROG)**

El nombre del archivo ejecutable de la aplicación, especificado como nombre de archivo totalmente calificado.

### **USRDATA('open, close, 235')**

Datos definidos por el usuario, que la aplicación puede utilizar.

## Visualizar la definición de proceso

El mandato **DSPMQMPCRC** le permite examinar los resultados de la definición. Por ejemplo:

```
MQMNAME(MYQUEUEMANAGER) DSPMQMPCRC('motor.insurance.quote.process')
```

También puede utilizar el mandato **CHGMQMPCRC** para modificar una definición de proceso existente, y el mandato **DLTMQMPCRC** para suprimir una definición de proceso.

## Comunicación entre dos sistemas IBM MQ en IBM i

Este ejemplo de código muestra cómo configurar dos sistemas IBM MQ for IBM i, utilizando mandatos CL, para que puedan comunicarse entre sí.

Los sistemas se llaman SYSTEMA y SYSTEMB, y se utiliza el protocolo de comunicaciones TCP/IP.



Lleve a cabo el procedimiento siguiente:

1. Cree un gestor de colas en SYSTEMA, y llámelo QMGRA1.

```
CRTMQM  MQMNAME(QMGRA1) TEXT('System A - Queue +
Manager 1') UDLMSGQ(SYSTEM.DEAD.LETTER.QUEUE)
```

2. Inicie este gestor de colas.

```
STRMQM  MQMNAME(QMGRA1)
```

3. Defina los objetos IBM MQ en SYSTEMA que necesita para enviar mensajes a un gestor de colas en SYSTEMB.

```
/* Transmission queue */
CRTMQMQ  QNAME(XMITQ.TO.QMGRB1) QTYPE(*LCL) +
MQMNAME(QMGRA1) TEXT('Transmission Queue +
to QMGRB1') MAXDEPTH(5000) USAGE(*TMQ)

/* Remote queue that points to a queue called TARGETB */
/* TARGETB belongs to queue manager QMGRB1 on SYSTEMB */
CRTMQMQ  QNAME(TARGETB.ON.QMGRB1) QTYPE(*RMT) +
MQMNAME(QMGRA1) TEXT('Remote Q pointing +
at Q TARGETB on QMGRB1 on Remote System +
SYSTEMB') RMTQNAME(TARGETB) +
RMTMQMNAME(QMGRB1) TMQNAME(XMITQ.TO.QMGRB1)

/* TCP/IP sender channel to send messages to the queue manager on SYSTEMB*/
CRTMQMCHL CHLNAME(QMGRA1.TO.QMGRB1) CHLTYPE(*SDR) +
MQMNAME(QMGRA1) TRPTYPE(*TCP) +
TEXT('Sender Channel From QMGRA1 on +
SYSTEMA to QMGRB1 on SYSTEMB') +
CONNNAME(SYSTEMB) TMQNAME(XMITQ.TO.QMGRB1)
```

4. Cree un gestor de colas en SYSTEMB, y llámelo QMGRB1.

```
CRTMQM  MQMNAME(QMGRB1) TEXT('System B - Queue +
Manager 1') UDLMSGQ(SYSTEM.DEAD.LETTER.QUEUE)
```

5. Inicie el gestor de colas en SYSTEMB.

```
STRMQM  MQMNAME(QMGRB1)
```

6. Defina los objetos IBM MQ que necesita para recibir mensajes del gestor de colas en SYSTEMA.

```
/* Local queue to receive messages on */
CRTMQMQ  QNAME(TARGETB) QTYPE(*LCL) MQMNAME(QMGRB1) +
TEXT('Sample Local Queue for QMGRB1')

/* Receiver channel of the same name as the sender channel on SYSTEMA */
CRTMQMCHL CHLNAME(QMGRA1.TO.QMGRB1) CHLTYPE(*RCVR) +
MQMNAME(QMGRB1) TRPTYPE(*TCP) +
TEXT('Receiver Channel from QMGRA1 to +
QMGRB1')
```

7. Por último, inicie un escucha TCP/IP en SYSTEMB para que se pueda iniciar el canal. En este ejemplo se utiliza el puerto predeterminado 1414.

```
STRMQMLSR MQMNAME(QMGRB1)
```

Ahora ya está preparado para enviar mensajes de prueba entre SYSTEMA y SYSTEMB. Utilizando uno de los ejemplos suministrados, ponga una serie de mensajes en la cola remota situada en SYSTEMA.

Inicie el canal en SYSTEMA, ya sea mediante el mandato **STRMQMCHL** o bien mediante el mandato **WRKMQMCHL** y entrando una solicitud de inicio (Opción 14) para el canal emisor.

El canal debe pasar al estado RUNNING (en ejecución) y los mensajes se envían a la cola TARGETB situada en SYSTEMB.

Para comprobar los mensajes, emita el mandato:

```
WRKMQMSG QNAME(TARGETB) MQMNAME(QMGRB1).
```

**IBM i**

## Definiciones de recurso de ejemplo en IBM i

Este ejemplo contiene el programa CL de ejemplo de IBM i AMQSAMP4.

```
/* **** */
/*
/* Program name: AMQSAMP4
/*
/* Description: Sample CL program defining MQM queues
/*           to use with the sample programs
/*           Can be run, with changes as needed, after
/*           starting the MQM
/*
/* <N_OCO_COPYRIGHT>
/* Licensed Materials - Property of IBM
/*
/* 63H9336
/* (c) Copyright IBM Corp. 1993, 2024. All Rights Reserved.
/*
/* US Government Users Restricted Rights - Use, duplication or
/* disclosure restricted by GSA ADP Schedule Contract with
/* IBM Corp.
/* <NOC_COPYRIGHT>
/*
/* **** */
/* Function:
/*
/* AMQSAMP4 is a sample CL program to create or reset the
/* MQI resources to use with the sample programs.
/*
/* This program, or a similar one, can be run when the MQM
/* is started - it creates the objects if missing, or resets
/* their attributes to the prescribed values.
/*
/*
/* Exceptions signaled: none
/* Exceptions monitored: none
/*
/* AMQSAMP4 takes a single parameter, the Queue Manager name
/*
/* **** */
/*SYS/PGM PARM(&QMGRNAME)

/* **** */
/* Queue Manager Name Parameter
/* **** */
/*SYS/DCL VAR(&QMGRNAME) TYPE(*CHAR)

/* **** */
/* EXAMPLES OF DIFFERENT QUEUE TYPES
/*
/* Create local, alias and remote queues
/*
/* Uses system defaults for most attributes
/*
/* **** */
/* Create a local queue */
CRTMQMQ QNAME('SYSTEM.SAMPLE.LOCAL') +
MQMNAME(&QMGRNAME) +
QTYPE(*LCL) REPLACE(*YES) +
+
TEXT('Sample local queue') /* description */+
SHARE(*YES) /* Shareable */+
DFTMSGPST(*YES) /* Persistent messages OK */
```

```

/* Create an alias queue */
CRTMQMQ QNAME('SYSTEM.SAMPLE.ALIAS')      +
MQMNAME(&QMGRNAME)                        +
QTYPE(*ALS) REPLACE(*YES)                 +
+
TEXT('Sample alias queue')                 +
DFTMSGPST(*YES) /* Persistent messages OK */+
TGTONAME('SYSTEM.SAMPLE.LOCAL')

/* Create a remote queue - in this case, an indirect reference */
/* is made to the sample local queue on OTHER queue manager */
CRTMQMQ QNAME('SYSTEM.SAMPLE.REMOTE')      +
MQMNAME(&QMGRNAME)                        +
QTYPE(*RMT) REPLACE(*YES)                 +
+
TEXT('Sample remote queue')/* description */+
DFTMSGPST(*YES) /* Persistent messages OK */+
RMTQNAME('SYSTEM.SAMPLE.LOCAL')           +
RMTMQMNAME(OTHER) /* Queue is on OTHER */

/* Create a transmission queue for messages to queues at OTHER */
/* By default, use remote node name */
CRTMQMQ QNAME('OTHER') /* transmission queue name */+
MQMNAME(&QMGRNAME)                        +
QTYPE(*LCL) REPLACE(*YES)                 +
TEXT('Transmission queue to OTHER') +
USAGE(*TMQ) /* transmission queue */

/*****
/* SPECIFIC QUEUES AND PROCESS USED BY SAMPLE PROGRAMS */
/* Create local queues used by sample programs */
/* Create MQI process associated with sample initiation queue */
*****/
/* General reply queue */
CRTMQMQ QNAME('SYSTEM.SAMPLE.REPLY')      +
MQMNAME(&QMGRNAME)                        +
QTYPE(*LCL) REPLACE(*YES)                 +
+
TEXT('General reply queue')                 +
DFTMSGPST(*NO) /* Not Persistent */

/* Queue used by AMQSINQ4 */
CRTMQMQ QNAME('SYSTEM.SAMPLE.INQ')        +
MQMNAME(&QMGRNAME)                        +
QTYPE(*LCL) REPLACE(*YES)                 +
+
TEXT('Queue for AMQSINQ4')                 +
SHARE(*YES) /* Shareable */+
DFTMSGPST(*NO) /* Not Persistent */+
+
TRGENBL(*YES) /* Trigger control on */+
TRGTYPE(*FIRST)/* Trigger on first message*/+
PRCNAME('SYSTEM.SAMPLE.INQPROCESS')       +
INITQNAME('SYSTEM.SAMPLE.TRIGGER')

/* Queue used by AMQSSET4 */
CRTMQMQ QNAME('SYSTEM.SAMPLE.SET')        +
MQMNAME(&QMGRNAME)                        +
QTYPE(*LCL) REPLACE(*YES)                 +
+
TEXT('Queue for AMQSSET4')                 +
SHARE(*YES) /* Shareable */+
DFTMSGPST(*NO)/* Not Persistent */+
+
TRGENBL(*YES) /* Trigger control on */+
TRGTYPE(*FIRST)/* Trigger on first message*/+
PRCNAME('SYSTEM.SAMPLE.SETPROCESS')       +
INITQNAME('SYSTEM.SAMPLE.TRIGGER')

/* Queue used by AMQSECH4 */
CRTMQMQ QNAME('SYSTEM.SAMPLE.ECHO')      +
MQMNAME(&QMGRNAME)                        +
QTYPE(*LCL) REPLACE(*YES)                 +
+
TEXT('Queue for AMQSECH4')                 +
SHARE(*YES) /* Shareable */+
DFTMSGPST(*NO)/* Not Persistent */+
+
TRGENBL(*YES) /* Trigger control on */+
TRGTYPE(*FIRST)/* Trigger on first message*/+

```

```

PRCNAME('SYSTEM.SAMPLE.ECHOPROCESS') +
INITQNAME('SYSTEM.SAMPLE.TRIGGER')

/* Initiation Queue used by AMQSTRG4, sample trigger process */
CRTMQMQ QNAME('SYSTEM.SAMPLE.TRIGGER') +
MQMNAME(&QMGRNAME) +
QTYPE(*LCL) REPLACE(*YES) +
TEXT('Trigger queue for sample programs')

/* MQI Processes associated with triggered sample programs */
/*
/***** Note - there are versions of the triggered samples *****/
/***** in different languages - set APPID for these *****/
/***** process to the variation you want to trigger *****/
/*
CRTMQMPRC PRCNAME('SYSTEM.SAMPLE.INQPROCESS') +
MQMNAME(&QMGRNAME) +
REPLACE(*YES) +
TEXT('Trigger process for AMQSINQ4') +
ENVDATA('JOBPTY(3)') /* Submit parameter */ +
/** Select the triggered program here **/ +
APPID('QMOM/AMQSINQ4') /* C */ +
/* APPID('QMOM/AMQOINQ4') /* COBOL */ +
/* APPID('QMOM/AMQ3INQ4') /* RPG - ILE */

CRTMQMPRC PRCNAME('SYSTEM.SAMPLE.SETPROCESS') +
MQMNAME(&QMGRNAME) +
REPLACE(*YES) +
TEXT('Trigger process for AMQSSET4') +
ENVDATA('JOBPTY(3)') /* Submit parameter */ +
/** Select the triggered program here **/ +
APPID('QMOM/AMQSSET4') /* C */ +
/* APPID('QMOM/AMQOSET4') /* COBOL */ +
/* APPID('QMOM/AMQ3SET4') /* RPG - ILE */

CRTMQMPRC PRCNAME('SYSTEM.SAMPLE.ECHOPROCESS') +
MQMNAME(&QMGRNAME) +
REPLACE(*YES) +
TEXT('Trigger process for AMQSECH4') +
ENVDATA('JOBPTY(3)') /* Submit parameter */ +
/** Select the triggered program here **/ +
APPID('QMOM/AMQSECH4') /* C */ +
/* APPID('QMOM/AMQOECH4') /* COBOL */ +
/* APPID('QMOM/AMQ3ECH4') /* RPG - ILE */

/*****
/*
/* Normal return.
/*
/*****
SNDPGMMSG MSG('AMQSAMP4 Completed creating sample +
objects for ' *CAT &QMGRNAME)
RETURN
ENDPGM

/*****
/*
/* END OF AMQSAMP4
/*
/*****

```

## IBM i Formas alternativas de administrar IBM MQ for IBM i

El uso de mandatos CL es el método preferido de administración de IBM MQ for IBM i. Sin embargo, puede utilizar otros métodos de administración, incluidos los mandatos MQSC, los mandatos PCF y la administración remota.

Normalmente se utilizan mandatos CL de IBM i para administrar IBM MQ for IBM i. Para obtener una visión general de estos mandatos, consulte [“Gestión de IBM MQ for IBM i utilizando mandatos CL”](#) en la [página 359](#).

Puede utilizar los sucesos de instrumentación de IBM MQ para supervisar el funcionamiento de los gestores de colas. Consulte [Sucesos de instrumentación](#) para obtener información sobre los sucesos de instrumentación de IBM MQ y cómo utilizarlos.

Se puede usar cualquier de los métodos de administración descritos en los subtemas siguientes como alternativa a los mandatos CL IBM i:

## Administración local y remota en IBM i

Los objetos de IBM MQ for IBM i se pueden administrar de forma local o remota.

La *administración local* consiste en llevar a cabo las tareas de administración en cualquier gestor de colas definido en el sistema local. En IBM MQ, esto puede considerarse como administración local porque no hay ningún canal IBM MQ implicado, es decir, la comunicación la gestiona el sistema operativo. Para realizar este tipo de tarea, hay que iniciar la sesión en el sistema remoto y ejecutar los mandatos desde allí o crear un proceso que pueda ejecutarlos automáticamente.

IBM MQ da soporte a la administración desde un único punto mediante lo que se conoce como *administración remota*. La administración remota consiste en enviar mensajes de control PCF (formato de mandato programable) a la cola SYSTEM.ADMIN.COMMAND.QUEUE situada en el gestor de colas de destino.

Hay varias maneras de generar mensajes PCF. Son las siguientes:

1. Escribir un programa que utilice mensajes PCF. Consulte el apartado [“Administración utilizando mandatos PCF en IBM i”](#) en la página 375.
2. Escribir un programa que utilice la MQAI, que envía los mensajes PCF. Consulte el apartado [“Utilizar la MQAI para simplificar el uso de los PCF”](#) en la página 37.
3. Utilizar IBM MQ Explorer, disponible con IBM MQ for Windows, que le permite utilizar una interfaz gráfica de usuario (GUI) y genera los mensajes PCF correctos. Consulte el apartado [“Utilización de IBM MQ Explorer con IBM MQ for IBM i”](#) en la página 375.
4. Utilizar **STRMQMQSC** para enviar mandatos indirectamente a un gestor de colas remoto. Consulte el apartado [“Administración con mandatos MQSC en IBM i”](#) en la página 373.

Por ejemplo, puede emitir un mandato remoto para cambiar una definición de cola en un gestor de colas remoto.

Algunos mandatos no se pueden emitir de esta manera, especialmente los que crean o inician gestores de colas y los que inician servidores de mandatos. Para realizar este tipo de tarea, hay que iniciar la sesión en el sistema remoto y ejecutar los mandatos desde allí o crear un proceso que pueda ejecutarlos automáticamente.

## Administración con mandatos MQSC en IBM i

Consulte este tema para obtener información sobre los mandatos MQSC y cómo utilizarlos para administrar IBM MQ for IBM i.

Los mandatos de script de IBM MQ (MQSC) se escriben en formato legible por personas, es decir, en texto EBCDIC. Utilice mandatos MQSC para gestionar objetos del gestor de colas, incluidos el propio gestor de colas, colas, definiciones de proceso, listas de nombres, canales, canales de conexión de cliente, escuchas, servicios, temas y objetos de información de autenticación.

Los mandatos MQSC se emiten a un gestor de colas utilizando el mandato CL **STRMQMQSC** IBM MQ. Éste es un método sólo de proceso por lotes, que toma su entrada de un archivo físico fuente en el sistema de bibliotecas de servidor. El nombre predeterminado para este archivo físico de origen es QMQSC.



**Atención:** No utilice la biblioteca QTEMP como biblioteca de origen para STRMQMQSC, ya que el uso de la biblioteca QTEMP es limitado. Debe utilizar otra biblioteca como un archivo de entrada para el mandato.

IBM MQ for IBM i no proporciona un archivo de origen denominado QMQSC. Para procesar mandatos MQSC, debe crear el archivo de origen QMQSC en una biblioteca de su elección, emitiendo el mandato siguiente:

```
CRTSRCPF FILE(MYLIB/QMQSC) RCDLEN(240) TEXT('IBM MQ - MQSC Source')
```

El código fuente de MQSC se guarda en los miembros de este archivo fuente. Para trabajar con los miembros, entre el siguiente mandato:

```
WRKMBRPDM MYLIB/QMQSC
```

Ahora podrá añadir miembros nuevos y mantener los existentes

También puede entrar mandatos MQSC interactivamente, emitiendo RUNMQSC, o bien:

1. Escriba el nombre del gestor de colas y pulse la tecla Enter para acceder al panel de resultados de **WRKMQM**.
2. Seleccionando F23=More options en este panel.
3. Seleccionando la opción 26 contra un gestor de colas activo en el panel que aparece en la [Figura 26](#) en la página 374.

Para finalizar una sesión MQSC de este tipo, escriba end.

La [Figura 26](#) en la [página 374](#) es un extracto de un archivo de mandatos MQSC que muestra un mandato MQSC (DEFINE QLOCAL) con sus atributos.

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

*Figura 26. Extracto del archivo de mandatos MQSC llamado myprog.in*

Por motivos de portabilidad entre los entornos IBM MQ, limite la longitud de línea en los archivos de mandatos MQSC a 72 caracteres. El signo más indica que el mandato continúa en la línea siguiente.

Los atributos de objeto especificados en MQSC aparecen en esta sección en mayúsculas (por ejemplo, RQMNAME), aunque no son sensibles a las mayúsculas y minúsculas.

**Nota:**

1. El formato de un archivo MQSC no depende de su ubicación en el sistema de archivos.
2. Los nombres de los atributos de MQSC pueden tener ocho caracteres como máximo.
3. Los mandatos MQSC están disponibles en otras plataformas, incluyendo z/OS.

Si desea una descripción de cada mandato MQSC y su sintaxis, consulte [Mandatos MQSC](#).

La finalidad de los mandatos PCF (formato de mandato programable) de IBM MQ es permitir que las tareas de administración se puedan programar en un programa de administración. De este modo, puede crear colas y definiciones de proceso y cambiar gestores de colas desde un programa.

Los mandatos PCF abarcan el mismo tipo de funciones que las proporcionadas por los mandatos MQSC. Sin embargo, a diferencia de los mandatos MQSC, los mandatos PCF y sus respuestas no están en un formato de texto legible para el usuario.

Puede escribir un programa que emita mandatos PCF a cualquier gestor de colas de la red desde un solo nodo. De este modo, puede centralizar y automatizar las tareas de administración.

Cada mandato PCF es una estructura de datos que se incluye en la parte de datos de aplicación de un mensaje de IBM MQ. El mandato se envía al gestor de colas de destino utilizando la función MQPUT de MQI, igual que cualquier otro mensaje. El servidor de mandatos situado en el gestor de colas que recibe el mensaje lo interpreta como un mensaje de mandato y ejecuta el mandato. Para obtener las respuestas, la aplicación emite una llamada MQGET y los datos de respuesta se devuelven en otra estructura de datos. La aplicación puede entonces procesar la respuesta y actuar en conformidad.

En resumen, estos son algunos de los elementos que un programador de aplicaciones debe especificar para crear un mensaje de mandato PCF:

#### **Descriptor de mensaje**

Es un descriptor de mensaje estándar de IBM MQ, en el que:

- El tipo de mensaje (*MsgType*) es MQMT\_REQUEST.
- El formato del mensaje (*Format*) es MQFMT\_ADMIN.

#### **Datos de aplicación**

Contienen el mensaje PCF, incluida la cabecera PCF, en el que:

- El tipo de mensaje PCF (*Type*) especifica MQCFT\_COMMAND.
- El identificador de mandato especifica el mandato, por ejemplo, *Change Queue* (MQCMD\_CHANGE\_Q).

Los PCF de escape son mandatos PCF que contienen mandatos MQSC dentro del texto del mensaje. Los PCF se pueden utilizar para enviar mandatos a un gestor de colas remoto. Consulte el apartado [“Utilizar la MQAI para simplificar el uso de los PCF”](#) en la página 37 para obtener más información.

Para obtener una descripción completa de las estructuras de datos PCF y cómo implementarlas, consulte [Estructuras para mandatos y respuestas](#).

Utilice esta información para administrar IBM MQ for IBM i con IBM MQ Explorer.

IBM MQ for Windows (plataforma x86) e IBM MQ para Linux (plataformas x86 y x86-64) proporcionan una interfaz de administración llamada IBM MQ Explorer para realizar tareas de administración, como alternativa al uso de mandatos CL, de control o MQSC.

IBM MQ Explorer le permite realizar la administración local o remota de la red desde un sistema que ejecute Windows (plataforma x86) o Linux (plataformas x86 y x86-64), apuntando IBM MQ Explorer a los gestores de colas y clústeres en los que esté interesado.

Con IBM MQ Explorer, puede hacer lo siguiente:

- Iniciar y detener un gestor de colas (sólo en la máquina local).
- Definir, visualizar y modificar las definiciones de objetos de IBM MQ, como colas, temas y canales.
- Examinar los mensajes de una cola.
- Iniciar y detener un canal.
- Ver información del estado de un canal.

- Ver los gestores de colas de un clúster.
- Comprobar qué aplicaciones, usuarios o canales tienen una determinada cola abierta.
- Crear un nuevo clúster de gestor de colas utilizando el asistente para **Crear un nuevo clúster**.
- Añadir un gestor de colas a un clúster utilizando el asistente para **Añadir un gestor de colas a un clúster**.
- Gestionar el objeto de información de autenticación, que se utiliza con la seguridad de canal de TLS (seguridad de la capa de transporte).

Utilizando la ayuda en línea, puede:

- Definir y controlar diversos recursos que incluyen gestores de colas, colas, canales, definiciones de proceso, canales de conexión de cliente, escuchas, servicios, listas de nombres y clústeres.
- Iniciar o detener un gestor de colas y los procesos asociados a él.
- Ver gestores de colas y sus objetos asociados en su estación de trabajo o desde otras estaciones de trabajo.
- Comprobar el estado de gestores de colas, clústeres y canales.

Asegúrese de haber satisfecho los siguientes requisitos antes de intentar utilizar IBM MQ Explorer para gestionar IBM MQ en una máquina servidor. Compruebe que:

1. Hay un servidor de mandatos en ejecución para cualquier gestor de colas que se esté administrando, iniciado en el servidor con el mandato CL **STRMQMCSVR**.
2. Existe un escucha TCP/IP adecuado para cada gestor de colas remoto. Este es el escucha de IBM MQ que se ha iniciado con el mandato **STRMQMLSR**.
3. El canal de conexión con el servidor, llamado SYSTEM.ADMIN.SVRCONN, existe en todos los gestores de colas remotos. Debe crear este canal usted mismo. Este canal es necesario para cada gestor de colas remoto que se esté administrando. Sin él, la administración remota no es posible.
4. Verifique que la cola SYSTEM.MQEXPLORER.REPLY.MODEL exista.

## **Gestión del servidor de mandatos para la administración remota en IBM i**

Utilice esta información sobre obtener más información sobre la administración remota del servidor de mandatos de IBM MQ for IBM i.

Cada gestor de colas puede tener un servidor de mandatos asociado. El servidor de mandatos procesa todos los mandatos entrantes procedentes de gestores de colas remotos o los mandatos PCF procedentes de aplicaciones. Presenta los mandatos al gestor de colas para que los procese y devuelve un código de terminación o un mensaje de operador, dependiendo del origen del mandato.

Es obligatorio tener un servidor de mandatos para toda tarea de administración que implique los PCF, la MQAI y también para la administración remota.

**Nota:** En la administración remota, debe asegurarse de que el gestor de colas de destino está ejecutándose. De lo contrario, los mensajes que contienen mandatos no pueden salir del gestor de colas desde el que se han emitido. En vez de ello, estos mensajes se transfieren a la cola de transmisión local que sirve al gestor de colas remoto. Evite esta situación en la medida de lo posible.

Existen mandatos de control individuales para iniciar y detener el servidor de mandatos. Puede realizar las operaciones descritas en las secciones siguientes utilizando IBM MQ Explorer.

### **Iniciar y detener el servidor de mandatos**

Para iniciar el servidor de mandatos, utilice este mandato CL:

```
STRMQMCSVR MQMNAME('saturn.queue.manager')
```

donde `saturn.queue.manager` es el gestor de colas para el que se inicia el servidor de mandatos.



Para detener el servidor de mandatos, utilice uno de los siguientes mandatos CL:

1. `ENDMQMSVR MQMNAME('saturn.queue.manager') OPTION(*CNTRLD)`

Realizar una detención controlada, donde `saturn.queue.manager` es el gestor de colas para el que el servidor de mandatos se está deteniendo. Esta es la opción predeterminada, lo que significa que `OPTION(*CNTRLD)` se puede omitir.

2. `ENDMQMSVR MQMNAME('saturn.queue.manager') OPTION(*IMMED)`

Realizar una detención inmediata, donde `saturn.queue.manager` es el gestor de colas para el que el servidor de mandatos se está deteniendo.

## Visualización del estado del servidor de mandatos

Para la administración remota, debe asegurarse de que el servidor de mandatos del gestor de colas de destino está ejecutándose. Si no se está ejecutando, no se podrá procesar ningún mandato remoto. Todos los mensajes que contienen mandatos se ponen en la cola de mandatos del gestor de colas de destino `SYSTEM.ADMIN.COMMAND.QUEUE`.

Para visualizar el estado del servidor de mandatos de un gestor de colas, llamado `saturn.queue.manager` en este ejemplo, el mandato CL es:

```
DSPMQMSVR MQMNAME('saturn.queue.manager')
```

Emita este mandato en la máquina destino. Si el servidor de mandatos está ejecutándose, aparece el panel de la [Figura 27 en la página 377](#):

```
Display MQM Command Server (DSPMQMSVR)

Queue manager name . . . . . > saturn.queue.manager
MQM Command Server Status. . . . > RUNNING

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

*Figura 27. Panel Visualizar servidor de mandatos de MQM*

## Ejecución de mandatos de consola web

Debe configurar el entorno tal como se describe en el texto siguiente, para que los mandatos de Qshell relacionados con la consola web se ejecuten correctamente en IBM MQ for IBM i.

### Acerca de esta tarea

Cuando se inicia Qshell, inicializa las tablas internas para procesar los mandatos basándose en el identificador de juego de caracteres codificados del trabajo. Para que los mandatos de Qshell relacionados con la consola web se ejecuten correctamente en IBM i, debe configurar el entorno.

Para establecer un entorno local, se establece la variable de entorno `LANG` en el nombre de vía de acceso de un objeto de entorno local. Por ejemplo, para establecer el entorno local para inglés de EE.UU., la variable de entorno `LANG` se establece de la siguiente manera:

```
LANG=/QSYS.LIB/EN_US.LOCALE
```

En el Qshell, puede comprobar el valor mediante el conjunto de mandatos de emisión para enumerar todas las variables de entorno. Normalmente, es LANG, que puede afectar al entorno local para el entorno de ejecución. También puede tener LC\_ALL.

Para ejecutar mandatos de Qshell correctamente, el valor del entorno local debe ser coherente con el valor del trabajo.

## Procedimiento

Utilice el mandato de CL DSPJOB JOB(JobNumber/USERProfile/JobName)

- a) Seleccione la opción 2 para mostrar los atributos de definición de trabajo.
- b) Los atributos siguientes deben ser coherentes con el valor de entorno LANG o LC\_ALL
  - Identificador de idioma
  - Identificador de país o región
  - Identificador de juego de caracteres codificado

Por ejemplo, si

```
LANG=/QSYS.LIB/FR_FR.LOCALE
```

los atributos de trabajo serían:

- ID de idioma . . . . . FRA
- ID de país o región . . . . . FR
- ID de juego de caracteres codificado. . . . . 297

## Qué hacer a continuación

Para obtener más información sobre el soporte de idioma nacional, consulte el siguiente tema IBM i en IBM Documentation: [Consideraciones sobre el soporte de idioma nacional \(NLS\)](#).

## IBM i Gestión de trabajo en IBM i

Esta información describe la forma en que IBM MQ maneja las solicitudes de trabajo y detalla las opciones disponibles para priorizar y controlar los trabajos asociados a IBM MQ.

### Aviso

No modifique los objetos de gestión de trabajo de IBM MQ a menos que comprenda perfectamente los conceptos de gestión de trabajo de IBM i e IBM MQ.

Puede encontrar información adicional sobre los subsistemas y las descripciones de trabajo en [Gestión de trabajo](#) en la documentación del producto IBM i. Preste especial atención a las secciones que tratan sobre el inicio y direccionamiento de trabajos ([Starting jobs](#) y [Batch jobs](#)).

IBM MQ for IBM i incorpora el entorno IBM i UNIX y hebras IBM i. **No** haga ningún cambio en los objetos del Sistema de archivos integrado (IFS).

Durante el funcionamiento normal, un gestor de colas de IBM MQ inicia una serie de trabajos por lotes para realizar diferentes tareas. De forma predeterminada, estos trabajos por lotes se ejecutan en el subsistema QMQM que se crea cuando se instala IBM MQ.

La gestión de trabajo se refiere al proceso de adaptar las tareas de IBM MQ para obtener el rendimiento óptimo del sistema, o para simplificar la administración.

Por ejemplo, puede:

- Cambiar la prioridad de ejecución de los trabajos para hacer que un gestor de colas responda antes que otro.
- Redirigir la salida de una serie de trabajos a una cola de salida concreta.
- Hacer que todos los trabajos de un cierto tipo se ejecuten en un subsistema específico.
- Aislar los errores en un subsistema.

La gestión de trabajo se lleva a cabo creando o cambiando las descripciones de trabajo asociadas a los trabajos de IBM MQ. Puede configurar la gestión de trabajo para:

- Toda una instalación de IBM MQ.
- Gestores de colas individuales.
- Trabajos individuales de gestores de colas individuales.

## IBM i Tareas IBM MQ para IBM i

Esta tabla contiene los trabajos de IBM MQ for IBM i y una breve descripción de cada uno.

Cuando un gestor de colas está en ejecución, puede ver algunos o todos los trabajos por lotes siguientes ejecutándose bajo el perfil de usuario QMQM en el subsistema de IBM MQ. Los trabajos se describen brevemente en la [Tabla 26 en la página 379](#).

Puede ver todos los trabajos conectados a un gestor de colas utilizando la opción 22 del panel **Trabajar con gestor de colas** (WRKMQM). Para ver los escuchas, utilice el mandato WRKMQLSR.

Nombre del trabajo	Función
AMQZMUC0	Gestor de programas de utilidad. Este trabajo ejecuta programas de utilidad del gestor de colas críticos, por ejemplo, el gestor de cadenas de diario.
AMQZXMA0	El controlador de ejecución que es el primer trabajo que inicia el gestor de colas. Maneja las solicitudes MQCONN e inicia procesos de agente para que procesen las llamadas de API de IBM MQ.
AMQZFUMA	Gestor de autorizaciones sobre objetos (OAM).
AMQZLAA0	Agentes de gestor de colas que realizan la mayor parte del trabajo de las aplicaciones que se conectan al gestor de colas utilizando MQCNO_STANDARD_BINDING.
AMQZLSA0	Agente de gestor de colas.
AMQZMUFO	Gestor de programas de utilidad
AMQZMGRO	Controlador de procesos. Este trabajo se utiliza para iniciar y gestionar escuchas y servicios.
AMQZMURO	Gestor de programas de utilidad. Este trabajo ejecuta programas de utilidad del gestor de colas críticos, por ejemplo, el gestor de cadenas de diario.
AMQFQPUB	Daemon de publicación/suscripción en cola.
AMQFCXBA	Trabajo de operador de intermediario.
RUNMQBRK	Trabajo de control de intermediario.
AMQRMPPA	Trabajo de agrupación de procesos de canal.
AMQCRSTA	Canal de respuesta invocado a través de TCP/IP.
AMQCRS6B	Canal receptor y conexión de cliente de LU62 (vea la nota).
AMQRRMFA	Gestor de depósitos para clústeres.
AMQCLMAA	Escucha TCP/IP sin hebras.

Tabla 26. Tareas de IBM MQ. (continuación)

Nombre del trabajo	Función
AMQPCSEA	Procesador de mandatos PCF que maneja las peticiones de administración remota y PCF.
RUNMQTRM	Supervisor desencadenante.
RUNMQDLQ	Manejador de cola de mensajes no entregados.
RUNMQCHI	El iniciador de canal.
RUNMQCHL	Trabajo de canal emisor que se inicia para cada canal emisor.
RUNMQLSR	Escucha TCP/IP con hebras.
AMQRCMLA	Procesador de mandatos MQSC y PCF de canal.

**Nota:** Se ejecuta el trabajo receptor LU62 en el subsistema de comunicaciones y adquiere sus propiedades de ejecución de las entradas de direccionamiento y de comunicaciones que se utilizan para iniciar el trabajo. Consulte [Extremo iniciado \(receptor\)](#) para obtener más información.

## IBM i Objetos de gestión de trabajo en IBM i

Cuando se instala IBM MQ, se proporcionan diversos objetos en la biblioteca QMQM para ayudarle con la gestión del trabajo. Estos objetos son los necesarios para que los trabajos de IBM MQ se ejecuten en el subsistema que les corresponde.

Se proporcionan descripciones de trabajo de ejemplo para dos de los trabajos por lotes de IBM MQ. Si no se proporciona ninguna descripción de trabajo específica para un trabajo de IBM MQ, el trabajo se ejecuta con la descripción de trabajo predeterminada QMQMJOB.

Los objetos de gestión de trabajo que se suministran cuando se instala IBM MQ se listan en la [Tabla 27 en la página 380](#) y los objetos que se crean para un gestor de colas se listan en la [Tabla 28 en la página 381](#).

**Nota:** Los objetos de gestión de trabajo pueden encontrarse en la biblioteca QMQM y los objetos del gestor de colas pueden encontrarse en la biblioteca del gestor de colas.

Tabla 27. Objetos de gestión de trabajo

Nombre	Tipo	Descripción
AMQZLAA0	*JOB	La descripción de trabajo que utilizan los procesos de agente de IBM MQ
AMQZLSA0	*JOB	El agente del gestor de colas de enlaces aislados
AMQZXMA0	*JOB	La descripción de trabajo que utilizan los controladores de ejecución de IBM MQ
QMQM	*SBS	El subsistema en el que se ejecutan todos los trabajos de IBM MQ
QMQM	*JOB	La cola de trabajos conectada al subsistema suministrado
QMQMJOB	*JOB	La descripción de trabajo predeterminada de IBM MQ, que utiliza si no hay una descripción de trabajo específica para un trabajo
QMQMMSG	*MSG	La cola de mensajes predeterminada para los trabajos de IBM MQ
QMQMRUN20	*CLS	Una descripción de clase para los trabajos de IBM MQ de prioridad alta

<i>Tabla 27. Objetos de gestión de trabajo (continuación)</i>		
<b>Nombre</b>	<b>Tipo</b>	<b>Descripción</b>
QMQRUN35	*CLS	Una descripción de clase para los trabajos de IBM MQ de prioridad media
QMQRUN50	*CLS	Una descripción de clase para los trabajos de IBM MQ de prioridad baja

<i>Tabla 28. Objetos de gestión de trabajo creados para un gestor de colas</i>		
<b>Nombre</b>	<b>Tipo</b>	<b>Descripción</b>
AMQA000000	*JRNRCV	Receptor de diario local
AMQAJRN	*JRN	Diario local
AMQJRNINF	*USRSPC	Espacio de usuario que se actualiza con los últimos receptores de diario necesarios para el arranque y la recuperación desde soporte de almacenamiento de un gestor de colas. Este espacio de usuario puede consultarlo una aplicación para determinar los receptores de diario que requieren archivado y los que se pueden suprimir de forma segura.
AMQAJRNMSG	*MSGQ	Cola de mensajes de diario local
AMQCRC6B	*PGM	Programa para iniciar la conexión LU6.2
AMQRFOLD	*XX_ENCODE_CASE_ONE archivo	Archivo de definiciones de canal del gestor de colas migrado
QMQMMSG	*MSGQ	Cola de mensajes del gestor de colas

## **Cómo IBM MQ usa los objetos de gestión de trabajo en IBM i**

Esta información describe la forma en que IBM MQ utiliza los objetos de gestión de trabajo, y proporciona ejemplos de configuración.



**Atención:** No modifique los valores de entrada de la cola de trabajos en el subsistema QMQM para limitar el número de trabajos permitidos en el subsistema según la prioridad. Si intenta hacer esto, puede impedir la ejecución de trabajos de IBM MQ esenciales tras su envío y hacer que el inicio del gestor de colas falle.

Para entender cómo se configura la gestión de trabajo, es necesario entender primero cómo utiliza IBM MQ la descripciones de trabajo.

La descripción de trabajo empleada para iniciar el trabajo controla numerosos atributos del trabajo. Por ejemplo:

- La cola de trabajos en la que se pone el trabajo y el subsistema en el que se ejecuta el trabajo.
- Los datos de direccionamiento que se utilizan para iniciar el trabajo y la clase que el trabajo emplea para los parámetros en tiempo de ejecución.
- La cola de salida que el trabajo utiliza para los archivos de impresión.

El proceso de iniciar un trabajo de IBM MQ puede considerarse un proceso de tres pasos:

1. IBM MQ selecciona una descripción de trabajo.

IBM MQ utiliza la siguiente técnica para determinar qué descripción de trabajo se ha de utilizar para un trabajo por lotes:

- a. Busca en la biblioteca del gestor de colas una descripción de trabajo que tenga el nombre del trabajo. Consulte [Comprender los nombres de biblioteca del gestor de colas IBM MQ for IBM i](#) para obtener más detalles sobre la biblioteca del gestor de colas.
  - b. Busca en la biblioteca del gestor de colas la descripción de trabajo predeterminada, que es QMQMJOB.
  - c. Busca en la biblioteca QMQM una descripción de trabajo cuyo nombre sea el del trabajo.
  - d. Utiliza la descripción de trabajo predeterminada, QMQMJOB, de la biblioteca QMQM.
2. El trabajo se somete a la cola de trabajos.

Las descripciones de trabajo que se suministran con IBM MQ se han configurado para que, de forma predeterminada, pongan los trabajos en la cola de trabajos QMQM de la biblioteca QMQM. La cola de trabajos QMQM está conectada al subsistema QMQM suministrado y, por eso, los trabajos empiezan a ejecutarse de forma predeterminada en el subsistema QMQM.

3. El trabajo entra en el subsistema y sigue los pasos de direccionamiento.

Cuando el trabajo entra en el subsistema, se utilizan los datos de direccionamiento indicados en la descripción de trabajo para localizar las entradas de direccionamiento del trabajo.

Los datos de direccionamiento deben corresponderse con una de las entradas de direccionamiento definidas en el subsistema QMQM, y ello define cuál de las clases suministradas (QMQRUN20, QMQRUN35 o QMQRUN50) va a utilizar el trabajo.

**Nota:** Si los trabajos de IBM MQ no parecen estar iniciándose, asegúrese de que el subsistema está ejecutándose y que la cola de trabajos no está retenida.

Si ha modificado los objetos de gestión de trabajo de IBM MQ, asegúrese de que todo está asociado correctamente. Por ejemplo, si especifica una cola de trabajos distinta de QMQM/QMQM en la descripción de trabajo, asegúrese de que se realiza un ADDJOBQE para el subsistema, que es QMQM.

Puede crear una descripción de trabajo para cada trabajo documentado en la [Tabla 26 en la página 379](#) utilizando la siguiente hoja de trabajo como ejemplo:

```
What is the queue manager library name? _____
Does job description AMQZXMA0 exist in the queue manager library? Yes No
Does job description QMQMJOB exist in the queue manager library? Yes No
Does job description AMQZXMA0 exist in the QMQM library? Yes No
Does job description QMQMJOB exist in the QMQM library? Yes No
```

Si responde No a todas estas preguntas, cree una descripción de trabajo global QMQMJOB en la biblioteca QMQM.

## La cola de mensajes de IBM MQ

Se crea una cola de mensajes de IBM MQ, QMQMMSG, en cada biblioteca de gestor de colas. Se envían mensajes del sistema operativo a esta cola cuando finalizan trabajos del gestor de colas y IBM MQ envía mensajes a la cola. Por ejemplo, para notificar qué receptores de diario se necesitan al arrancar. Procure mantener el número de mensajes de esta cola de mensajes en un tamaño razonable para que sea más fácil de supervisar.

## Ejemplos de funcionamiento predeterminado en IBM i

Estos ejemplos muestran cómo funciona una instalación de IBM MQ sin modificar cuando algunos de los trabajos estándar se someten durante el inicio del gestor de colas.

En primer lugar, se inicia el trabajo del controlador de ejecución AMQZXMA0.

1. Emita el mandato **STRMQM** para el gestor de colas TESTQM.
2. IBM MQ realiza una búsqueda en la biblioteca de gestor de colas QMTESTQM, primero de la descripción de trabajo AMQZXMA0 y luego de la descripción de trabajo QMQMJOB.

No existe ninguna de estas descripciones de trabajo, por lo que IBM MQ busca la descripción de trabajo AMQZXMA0 en la biblioteca del producto QMQM. Esta descripción de trabajo existe, por lo que se emplea para someter el trabajo.

3. La descripción de trabajo utiliza la cola de trabajos predeterminada de IBM MQ, por lo que el trabajo se somete a la cola de trabajos QMQM/QMQM.
4. Los datos de direccionamiento existentes en la descripción de trabajo AMQZXMA0 son QMQMRUN20, de modo que el sistema busca en las entradas de direccionamiento del subsistema una entrada que se corresponda con esos datos.

De forma predeterminada, la entrada de direccionamiento que tiene el número de secuencia 9900 tiene datos de comparación que se corresponden con QMQMRUN20, por lo que el trabajo se inicia con la clase definida en esa entrada de direccionamiento, que también se llama QMQMRUN20.

5. La clase QMQM/QMQMRUN20 tiene la prioridad de ejecución establecida en 20; así que el trabajo AMQZXMA0 se ejecuta en el subsistema QMQM con la prioridad que tienen los trabajos más interactivos del sistema.

## IBM i Configuración de ejemplos de gestión de trabajo en IBM i

Utilice esta información para obtener más información acerca de cómo puede cambiar y crear descripciones de trabajo de IBM MQ para modificar los atributos de tiempo de ejecución de los trabajos de IBM MQ.

La clave de la flexibilidad de la gestión de trabajo de IBM MQ consiste en que IBM MQ busca las descripciones de trabajo en dos niveles:

- Si crea o cambia las descripciones de trabajo en una biblioteca del gestor de colas, esos cambios alteran temporalmente las descripciones de trabajo globales existentes en QMQM, pero los cambios son locales y solo afectan a ese gestor de colas en concreto.
- Si crea o cambia descripciones de trabajo globales en la biblioteca QMQM, esas descripciones de trabajo afectan a todos los gestores de colas del sistema, a menos que se alteren temporalmente de forma local a nivel de gestores de colas individuales.

1. El ejemplo siguiente muestra cómo aumentar la prioridad de los trabajos de control de canal para un gestor de colas individual.

Para lograr que los trabajos de gestor de repositorios y de iniciador de canal, llamados AMQRRMFA y RUNMQCHI, se ejecuten lo antes posible para el gestor de colas TESTQM, siga estos pasos:

- a. Cree duplicados locales de la descripción de trabajo QMQM/QMQMJOBDB con los nombres de los procesos de IBM MQ que desea controlar en la biblioteca del gestor de colas. Por ejemplo:

```
CRTDUPOBJ OBJ(QMQMJOBDB) FROMLIB(QMQM) OBJTYPE(*JOBDB) TOLIB(QMTESTQM)
NEWOBJ (RUNMQCHI)
CRTDUPOBJ OBJ(QMQMJOBDB) FROMLIB(QMQM) OBJTYPE(*JOBDB) TOLIB(QMTESTQM)
NEWOBJ (AMQRRMFA)
```

- b. Cambie el parámetro de datos de direccionamiento existente en la descripción de trabajo para asegurar que los trabajos utilicen la clase QMQMRUN20.

```
CHGJOBDB JOBDB(QMTESTQM/RUNMQCHI) RTGDTA('QMQMRUN20')
CHGJOBDB JOBDB(QMTESTQM/AMQRRMFA) RTGDTA('QMQMRUN20')
```

Ahora, los trabajos AMQRRMFA y RUNMQCHI del gestor de colas TESTQM van a:

- Utilizar las nuevas descripciones de trabajo locales que hay en la biblioteca del gestor de colas
  - Ejecutarse con la prioridad 20, pues se emplea la clase QMQMRUN20 cuando los trabajos entran en el subsistema
2. El ejemplo siguiente define una nueva clase de prioridad de ejecución para el subsistema QMQM.

- a. Cree una clase duplicada en la biblioteca QMQM para permitir que otros gestores de colas accedan a la clase, emitiendo el siguiente mandato:

```
CRTDUPOBJ OBJ(QMQMRUN20) FROMLIB(QMQM) OBJTYPE(*CLS) TOLIB(QMQM)
NEWOBJ(QMQMRUN10)
```

- b. Cambie la clase para que tenga la nueva prioridad de ejecución, emitiendo el siguiente mandato:

```
CHGCLS CLS(QMQM/QMQMRUN10) RUNPTY(10)
```

- c. Añada la nueva definición de clase en el subsistema, emitiendo el siguiente mandato:

```
ADDRTGE SBSDB(QMQM/QMQM) SEQNBR(8999) CMPVAL('QMQMRUN10') PGM(QSYS/QCMD)
CLS(QMQM/QMQMRUN10)
```

**Nota:** Puede especificar cualquier valor numérico para el número de secuencia de direccionamiento, pero los valores deben estar en orden secuencial. Este número de secuencia indica al subsistema el orden en que debe buscarse una coincidencia de los datos de direccionamiento en las entradas de direccionamiento.

- d. Cambie la descripción de trabajo local o global para que utilice la nueva clase de prioridad, emitiendo el siguiente mandato:

```
CHGJOB JOB(QMQMLibname/QMQMJOB) RTGDTA('QMQMRUN10')
```

Ahora todos los trabajos del gestor de colas asociados a QMLibraryname utilizan una prioridad de ejecución de 10.

3. El ejemplo siguiente enseña cómo se ejecuta un gestor de colas en su propio subsistema.

Para hacer que todos los trabajos del gestor de colas TESTQM se ejecuten en el subsistema QBATCH, siga estos pasos:

- a. Cree un duplicado local de la descripción de trabajo QMQM/QMQMJOB en la biblioteca del gestor de colas, con este mandato

```
CRTDUPOBJ OBJ(QMQMJOB) FROMLIB(QMQM) OBJTYPE(*JOB) TOLIB(QMTESTQM)
```

- b. Cambie el parámetro cola de trabajos, en la descripción de trabajo, para asegurar que los trabajos utilicen la cola de trabajos QBATCH.

```
CHGJOB JOB(QMTESTQM/QMQMJOB) JOBQ(*LIBL/QBATCH)
```

**Nota:** La cola de trabajos está asociada a la descripción del subsistema. Si encuentra que los trabajos permanecen en la cola de trabajos, verifique que la definición de la cola de trabajos está definida en el SBSDB. Utilice el mandato DSPSBSDB para el subsistema y seleccione la opción 6, Entradas de la cola de trabajos.

Ahora, todos los trabajos del gestor de colas TESTQM van a:

- Utilizar la nueva descripción de trabajo local predeterminada que hay en la biblioteca del gestor de colas
- Enviarse a la cola de trabajos QBATCH

Para estar seguro de que los trabajos se direccionan correctamente y adquieren la prioridad deseada, puede:

- Crear entradas de direccionamiento para los trabajos de IBM MQ en el subsistema QBATCH, o bien
- Basarse en una entrada de direccionamiento global que invoque QCMD, sin tener en cuenta los datos de direccionamiento utilizados.



Esta opción sólo funciona si se establece el valor \*NOMAX en la opción de máximo de trabajos activos para la cola de trabajos QBATCH. El valor predeterminado del sistema es 1.

#### 4. El siguiente ejemplo crea otro subsistema IBM MQ

- a. Cree un subsistema duplicado en la biblioteca QMQM, emitiendo el siguiente mandato:

```
CRTDUPOBJ OBJ(QMQM) FROMLIB(QMQM) OBJTYPE(*SBSD) TOLIB(QMQM) NEWOBJ(QMQM2)
```

- b. Elimine la cola de trabajos QMQM, emitiendo el siguiente mandato:

```
RMVJOBQE SBSM(QMQM/QMQM2) JOBQ(QMQM/QMQM)
```

- c. Cree una nueva cola de trabajos para el subsistema, emitiendo el siguiente mandato:

```
CRTJOBQ JOBQ(QMQM/QMQM2) TEXT('Job queue for IBM MQ Queue Manager')
```

- d. Añada una entrada de la cola de trabajos en el subsistema, emitiendo el siguiente mandato:

```
ADDJOBQE SBSM(QMQM/QMQM2) JOBQ(QMQM/QMQM2) MAXACT(*NOMAX)
```

- e. Cree un QMQMJOBM duplicado en la biblioteca del gestor de colas, emitiendo el siguiente mandato:

```
CRTDUPOBJ OBJ(QMQMJOBM) FROMLIB(QMQM) OBJTYPE(*JOBM) TOLIB(QMlibraryname)
```

- f. Cambie la descripción de trabajo para que utilice la nueva cola de trabajos, emitiendo el siguiente mandato:

```
CHGJOBM JOBM(QMlibraryname/QMQMJOBM) JOBQ(QMQM/QMQM2)
```

- g. Inicie el subsistema emitiendo el siguiente mandato:

```
STRSBS SBSM(QMQM/QMQM2)
```

#### Nota:

- Puede especificar el subsistema en cualquier biblioteca. Si por algún motivo se vuelve a instalar el producto, o se sustituye la biblioteca QMQM, se eliminarán los cambios que haya realizado.
- Ahora todos los trabajos del gestor de colas asociados a QMlibraryname se ejecutan bajo el subsistema QMQM2.



## Disponibilidad, copia de seguridad, recuperación y reinicio en IBM i

Utilice esta información para entender cómo IBM MQ for IBM i utiliza el soporte de registro por diario de IBM i como ayuda para su estrategia de copia de seguridad y restauración.

Debe conocer los métodos de copia de seguridad y recuperación estándar de IBM i, así como con el uso de diarios y los receptores de diario asociados en IBM i antes de leer esta sección. Para obtener información sobre estos temas, consulte [Copia de seguridad y recuperación](#).

Para entender la estrategia de copia de seguridad y recuperación, primero debe entender cómo IBM MQ for IBM i organiza los datos en el sistema de archivos de IBM i y en el sistema de archivos integrado (IFS).

IBM MQ for IBM i guarda los datos en una biblioteca individual para cada instancia del gestor de colas y en archivos continuos del sistema de archivos IFS.

Las bibliotecas específicas de gestor de colas contienen diarios, receptores de diario y los objetos necesarios para controlar la gestión de trabajo del gestor de colas. Los directorios y archivos del IFS

contienen los archivos de configuración de IBM MQ, las descripciones de los objetos de IBM MQ y los datos contenidos en ellos.

Todo cambio que se realice en estos objetos (que sea recuperable en caso de anomalía del sistema) se registra en un diario *antes* de ser aplicado al correspondiente objeto. Así, estos cambios se pueden recuperar reproduciendo la información registrada en el diario.

Puede configurar IBM MQ for IBM i para que utilice varias instancias del gestor de colas en distintos servidores para aumentar la disponibilidad del gestor de colas y acelerar la recuperación en caso de anomalía de un servidor o del gestor de colas.

## IBM i Diarios del gestor de colas en IBM i

Utilice esta información para entender cómo IBM MQ for IBM i utiliza los diarios en su operación para controlar las actualizaciones realizadas en objetos locales.

Cada biblioteca de gestor de colas contiene un diario para ese gestor de colas, y el diario tiene el nombre QM *GRLIB*/AMQ *A* JRN, donde QM *GRLIB* es el nombre de la biblioteca del gestor de colas, y *A* es una letra, *A* en el caso de un gestor de colas de instancia única, que es exclusiva de la instancia del gestor de colas.

QM *GRLIB* toma el nombre QM, seguido del nombre del gestor de colas en un formato exclusivo. Por ejemplo, un gestor de colas llamado TEST tiene una biblioteca de gestor de colas llamado QMTEST. La biblioteca del gestor de colas se puede especificar al crear un gestor de colas mediante el mandato **CRTMQM**.

Los diarios tienen receptores de diario asociados, que contienen la información que se registra por diario. Los receptores son objetos a los que sólo se puede añadir información y que, con el tiempo, se llenan.

Los receptores de diario utilizan un valioso espacio de memoria con información anticuada. Sin embargo, para minimizar este problema, siempre puede colocar la información en un almacenamiento permanente. En cualquier momento, hay un receptor de diario conectado al diario. El receptor de diario, cuando llega al tamaño umbral predeterminado, se desconecta y se sustituye por un nuevo receptor de diario. Puede especificar el umbral de receptores de diario al crear un gestor de colas utilizando **CRTMQM** y el parámetro **THRESHOLD**.

Los receptores de diario asociados al diario local de IBM MQ para IBM i existen en cada biblioteca de gestor de colas y adoptan el siguiente convenio de denominación:

```
AMQ Arnnnnn
```

donde

**A**

es una letra de la A-Z. La A es para gestores de cola con una sola instancia. Variará para las distintas instancias de un gestor de colas multiinstancia.

**NNNN**

es 00000 to 99999 decimal que se incrementa en 1 para el siguiente diario de la secuencia.

**r**

es decimal 0 to 9, que se incrementa en 1 cada vez que se restaura un receptor.

La secuencia de los diarios se basa en la fecha. No obstante, la denominación del diario siguiente se basa en estas reglas:

1. A AMQA $r$ nnnnn le sigue AMQA $r$ (nnnnn+1) y el número nnnnn se recorta cuando llega a 99999. Por ejemplo, después de AMQA099999 viene AMQA000000 y después de AMQA999999 sigue AMQA900000.
2. Si ya existe un diario cuyo nombre se haya generado mediante la regla 1, se envía el mensaje CPI70E3 a la cola de mensajes QSYSOPR y se detiene la conmutación automática de receptores.

Se sigue utilizando el receptor conectado en ese momento hasta que se investigue el problema y se conecte manualmente un nuevo receptor.

3. Si se terminan los nombres de la secuencia (es decir, en el sistema ya existen todos los nombres de diario posibles), deberá realizar las dos tareas siguientes:
  - a. Suprimir los diarios que ya no se necesiten (consulte el apartado [“Gestión de diarios en IBM i”](#) en la página 391).
  - b. Registrar los cambios de diario en el receptor de diario más reciente (utilizando **RCDMQMIMG**) y luego repetir el paso anterior. Así se podrán reutilizar los nombres de los receptores de diario antiguos.

El diario AMQAJRN utiliza la opción MNGRCV(\*SYSTEM) para permitir que el sistema operativo cambie automáticamente los receptores de diario cuando se alcance el umbral. Para obtener más información sobre el modo en que el sistema maneja los receptores, consulte *IBM i Backup and Recovery*.

El valor umbral predeterminado del receptor de diario es de 100.000 KB. Al crear el gestor de colas, puede establecer este valor en uno más grande. El valor inicial del atributo LogReceiverSize se escribe en la stanza LogDefaults del archivo mqs.ini.

Cuando un receptor de diario se amplía más allá del umbral especificado, el receptor se separa y se crea un nuevo receptor de diario que hereda los atributos del receptor anterior. Los cambios en los atributos LogReceiverSize o LogASP después de que se haya creado un gestor de colas se ignoran cuando el sistema conecta automáticamente un nuevo receptor de diario.

Consulte [Modificación de la información de configuración en IBM i](#) para obtener más información sobre la configuración del sistema.

Si necesita cambiar el tamaño de los receptores de diario una vez creado el gestor de colas, cree un nuevo receptor de diario y establezca que su propietario es QMQM, utilizando para ello los siguientes mandatos:

```
CRTJRNRCV JRNRCV(QM GRLIB/AMQ Arnnnnn) THRESHOLD(xxxxxx) +
TEXT('MQM LOCAL JOURNAL RECEIVER')
CHGOBJOWN OBJ(QM GRLIB/AMQ Arnnnnn) OBJTYPE(*JRNRCV) NEWOWN(QMQM)
```

donde

**QMGRLIB**

Es el nombre de la biblioteca del gestor de colas

**A**

Es el identificador de la instancia (normalmente A).

**rnnnnn**

Es el siguiente receptor de diario en la secuencia de denominación descrita anteriormente

**xxxxxx**

Es el nuevo umbral de receptor (en KB)

**Nota:** El tamaño máximo del receptor está regido por el sistema operativo. Para comprobar este valor, mire la palabra clave THRESHOLD en el mandato **CRTJRNRCV**.

Ahora conecte el nuevo receptor al diario AMQAJRN con el mandato:

```
CHGJRN JRN(QMGRLIB/AMQ A JRN) JRNRCV(QMGRLIB/AMQ Annnnnn)
```

En [“Gestión de diarios en IBM i”](#) en la página 391 hallará los detalles de cómo puede gestionar estos receptores de diario.

**IBM i** *Uso del diario del gestor de colas en IBM i*

Utilice esta información para entender cómo IBM MQ for IBM i utiliza los diarios en su operación para controlar las actualizaciones realizadas en objetos locales.

Las actualizaciones persistentes realizadas en las colas de mensajes se producen en dos etapas. Primero, los registros que representan la actualización se graban en el diario y después se actualiza el archivo de cola.

Por lo tanto, puede suceder que los receptores de diario estén más actualizados que los archivos de las colas. Para asegurar que el proceso de reinicio empiece desde un punto coherente, IBM MQ utiliza puntos de comprobación.

Un punto de comprobación es un momento específico en el que el registro descrito en el diario es igual que el registro de la cola. El punto de comprobación propiamente dicho consta de una serie de registros de diario necesarios para reiniciar el gestor de colas. Por ejemplo, el estado de todas las transacciones (es decir, las unidades de trabajo) activas en el momento del punto de comprobación.

Los puntos de comprobación los genera automáticamente IBM MQ. Estos se toman cuando el gestor de colas se inicia y se cierra, y después de que se anoten un determinado número de operaciones.

Puede forzar que un gestor de colas tome un punto de comprobación emitiendo el mandato RCDMQMIMG en todos los objetos de un gestor de colas y visualizando los resultados, como se indica a continuación:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(Q_MGR_NAME) DSPJRNDTA(*YES)
```

A medida que las colas manejan nuevos mensajes, el registro de punto de comprobación deja de ser coherente con el estado actual de las colas.

Cuando se reinicia IBM MQ, éste localiza el registro de punto de comprobación más reciente en el registro. Esta información se guarda en el archivo de punto de comprobación que se actualiza al final de cada punto de comprobación. El registro de punto de comprobación representa el punto más reciente de coherencia entre el archivo de registro y los datos. Los datos de este punto de comprobación se utilizan para reconstruir las colas tal como eran en el momento del punto de comprobación. Al reconstruir las colas, se reproduce el registro para devolver a las colas el estado que tenían antes de la anomalía o el cierre del sistema.

Para entender cómo IBM MQ utiliza el diario, considere el caso de una cola local llamada TESTQ en el gestor de colas TEST. Esto se representa mediante el archivo de IFS:

```
/QIBM/UserData/mqm/qmgrs/TEST/queues
```

Si un mensaje especificado se transfiere a esta cola y luego se recupera de la cola, las acciones que tienen lugar se muestran en la [Figura 28](#) en la página 388.

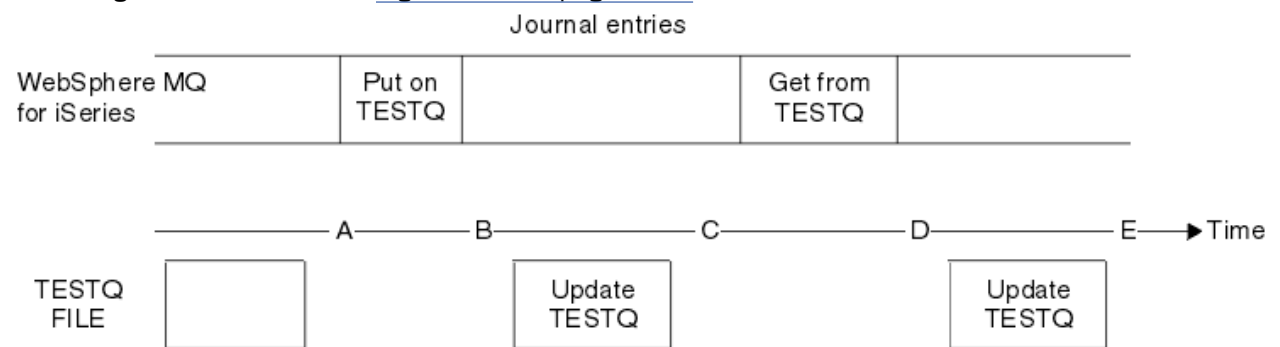


Figura 28. Secuencia de sucesos que se producen al actualizar objetos de MQM

Los cinco puntos, del A al E, que se muestran en el diagrama representan momentos que definen los siguientes estados:

- A** La representación del archivo de IFS de la cola es coherente con la información contenida en el diario.
- B** Se graba una entrada de diario en el diario que define una operación Put en la cola.
- C** Se realiza la correspondiente actualización en la cola.

## D

Se graba una entrada de diario en el diario que define una operación Get de la cola.

## E

Se realiza la correspondiente actualización en la cola.

La clave de las posibilidades de recuperación de IBM MQ for IBM i es que el usuario puede guardar la representación del archivo IFS de TESTQ como es en el momento A y, posteriormente, recuperar la representación del archivo IFS de TESTQ como es en el momento E, restaurando el objeto guardado y reproduciendo las entradas del diario a partir del momento A.

IBM MQ for IBM i utiliza esta estrategia para recuperar los mensajes persistentes después de una anomalía del sistema. IBM MQ recuerda una entrada determinada de los receptores de diario y garantiza que en el momento del inicio reproducirá las entradas de los diarios a partir de ese punto. Esta entrada de inicio se recalcula periódicamente para que IBM MQ sólo tenga que realizar la reproducción mínima necesaria en el próximo inicio.

IBM MQ proporciona recuperación individual de objetos. Toda la información persistente relacionada con un objeto se registra en los diarios locales de IBM MQ for IBM i. Cualquier objeto de IBM MQ que quede dañado o deteriorado se puede reconstruir completamente a partir de la información guardada en el diario.

Para obtener más información sobre el modo en que el sistema maneja los receptores, consulte [“Disponibilidad, copia de seguridad, recuperación y reinicio en IBM i”](#) en la página 385.

## IBM i **Imágenes de soporte en IBM i**

En IBM i, una imagen de soporte es una copia completa de un objeto IBM MQ que se registra en el diario. Algunos objetos corrompidos o dañados se pueden recuperar de forma automática a partir de su imagen de soporte.

Un objeto de IBM MQ de larga duración puede representar un gran número de entradas de diario, que se remontan al momento de su creación. Para evitarlo, existe en IBM MQ for IBM i el concepto de imagen de soporte de un objeto.

Esta imagen de soporte es una copia completa del objeto de IBM MQ registrado en el diario. Si se toma una imagen de un objeto, el objeto se puede reconstruir reproduciendo las entradas de diario desde la imagen en adelante. La entrada de diario que representa el punto de reproducción de cada objeto de IBM MQ se conoce como entrada de recuperación de soporte. IBM MQ hace un seguimiento de lo siguiente:

- La entrada de recuperación desde soporte para cada objeto del gestor de colas.
- La entrada más antigua de este conjunto (consulte el mensaje de error AMQ7462 de [“Gestión de diarios en IBM i”](#) en la [página 391](#) para obtener más detalles).

Periódicamente se toman imágenes del objeto \*CTLG y del objeto \*MQM, ya que estos objetos son esenciales para el reinicio del gestor de colas.

Las imágenes de los demás objetos se toman cuando sea conveniente. De forma predeterminada, se toman imágenes de todos los objetos cuando se cierra un gestor de colas usando el mandato **ENDMQM** con el parámetro ENDCCTJOB(\*YES). Esta operación puede requerir una cantidad de tiempo considerable si se trata de gestores de colas muy grandes. Si necesita cerrarlo rápidamente, especifique el parámetro RCDMQMIMG(\*NO) con ENDCCTJOB(\*YES). En estos casos, es recomendable registrar una imagen de soporte completa en los diarios después de que se haya reiniciado el gestor de colas, mediante el siguiente mandato:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(Q_MGR_NAME)
```

IBM MQ registra automáticamente una imagen de un objeto si encuentra un punto adecuado en el que un objeto se pueda describir de forma compacta realizando una pequeña entrada en el diario. Sin embargo, puede que esto no suceda nunca para algunos objetos, como son las colas que contienen siempre grandes cantidades de mensajes.

En lugar de permitir que la fecha de la entrada de recuperación de soporte más antigua continúe durante un periodo innecesariamente largo, utilice el mandato IBM MQ RCDMQMIMG, que le permite tomar una imagen de los objetos seleccionados manualmente.

## Recuperación a partir de imágenes de soporte

IBM MQ recupera automáticamente algunos objetos a partir de su imagen de soporte si se detecta que los objetos están dañados o deteriorados. En particular, esto se aplica a los objetos especiales \*MQM y \*CTLG como parte del arranque normal del gestor de colas. Si en el momento del último cierre del gestor de colas quedó incompleta alguna transacción de punto de sincronismo, también se recupera automáticamente cualquier cola afectada, con el fin de completar la operación de arranque.

Debe recuperar otros objetos manualmente, utilizando el mandato RCRMQMOBJ de IBM MQ. Este mandato reproduce las entradas del diario para volver a crear el objeto de IBM MQ. Si un objeto de IBM MQ resulta dañado, las únicas acciones válidas son suprimirlo o crearlo de nuevo con este método. Sin embargo, tenga en cuenta que los mensajes no persistentes no se pueden recuperar de este modo.

### IBM i **Puntos de comprobación en IBM MQ for IBM i**

Los puntos de comprobación se toman a diferentes horas para proporcionar un punto de inicio coherente y conocido para la recuperación.

La hebra de punto de comprobación en el proceso AMQZMUC0 se encarga de tomar el punto de comprobación en los puntos siguientes:

- Arranque del gestor de colas (STRMQM).
- Cierre del gestor de colas (ENDMQM).
- Cuando haya transcurrido una cantidad de tiempo desde el último punto de comprobación (el periodo predeterminado son 30 minutos) y se haya grabado un número mínimo de registros de anotaciones desde el último punto de comprobación (el valor predeterminado es 100).
- Después de que se haya grabado un número de registros de anotaciones. El valor predeterminado es 10 000.
- Después de que se haya sobrepasado el tamaño umbral de diario y se haya creado un receptor de nuevo diario.
- Cuando se toma una imagen de soporte con:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(Q_MGR_NAME) DSPJRNDTA(*YES)
```

### IBM i **Copias de seguridad de datos de IBM MQ for IBM i**

Utilice esta información para conocer los dos tipos de copias de seguridad de IBM MQ para cada gestor de colas.

Para cada gestor de colas, hay dos tipos de copia de seguridad de IBM MQ a tener en cuenta:

- Copia de seguridad de datos y de diario.

Para asegurar que ambos tipos de datos sean compatibles, haga esta copia de seguridad únicamente después de cerrar el gestor de colas.

- Copia de seguridad de diario.

Puede hacerse mientras el gestor de colas está activo.

Para ambos métodos, debe encontrar los nombres del directorio IFS del gestor de colas y de la biblioteca del gestor de colas. Puede encontrarlos en el archivo de configuración de IBM MQ (mq5.ini). Para obtener más información, consulte [La stanza QueueManager](#).

Utilice los procedimientos siguientes para realizar ambos tipos de copia de seguridad:

## Copia de seguridad de datos y de diario de un determinado gestor de colas

**Nota: No utilice una solicitud de guardar-mientras-está-activo cuando el gestor de colas esté ejecutándose. No se puede completar una solicitud de este tipo a menos que se hayan confirmado o retrotraído todas las definiciones de compromiso que tengan cambios pendientes. Si se utiliza este mandato cuando el gestor de colas está activo, las conexiones de los canales podrían no finalizar con normalidad. Utilice siempre el procedimiento siguiente.**

1. Cree un receptor de diario vacío, utilizando el siguiente mandato:

```
CHGJRN JRN(QMTEST/AMQAJRN) JRNRCV(*GEN)
```

2. Utilice el mandato **RCDMQMIMG** para registrar una imagen MQM para todos los objetos IBM MQ y, a continuación, fuerce un punto de comprobación utilizando el mandato:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) DSPJRNDTA(*YES) MQMNAME(TEST)
```

3. Finalice los canales y asegúrese de que el gestor de colas no está ejecutándose. Si el gestor de colas se está ejecutando, deténgalo con el mandato **ENDMQM**.
4. Haga copia de seguridad de la biblioteca del gestor de colas emitiendo el siguiente mandato:

```
SAVLIB LIB(QMTEST)
```

5. Haga copia de seguridad de los directorios de IFS del gestor de colas emitiendo el siguiente mandato:

```
SAV DEV(...) OBJ(('QIBM/UserData/mqm/qmgrs/test'))
```

## Copia de seguridad de diario de un determinado gestor de colas

En el supuesto de que en algún momento haya hecho una operación de guardar completa y debido a que toda la información relevante se guarda en los diarios, puede realizar copias de seguridad parciales guardando los receptores de diario. Estas registran todos los cambios realizados a partir del momento de la copia de seguridad completa y, para ello, se emiten los siguientes mandatos:

1. Cree un receptor de diario vacío, utilizando el siguiente mandato:

```
CHGJRN JRN(QMTEST/AMQAJRN) JRNRCV(*GEN)
```

2. Utilice el mandato **RCDMQMIMG** para registrar una imagen MQM para todos los objetos IBM MQ y, a continuación, fuerce un punto de comprobación utilizando el mandato:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) DSPJRNDTA(*YES) MQMNAME(TEST)
```

3. Guarde los receptores de diario utilizando el siguiente mandato:

```
SAVOBJ OBJ(AMQ*) LIB(QMTEST) OBJTYPE(*JRNRCV) .....
```

Una estrategia de copia de seguridad sencilla consiste en realizar todas las semanas una copia de seguridad completa de las bibliotecas de IBM MQ y hacer todos los días una copia de seguridad de los diarios. Esto, como es natural, depende de cómo haya configurado la estrategia de copia de seguridad de la empresa.

## **Gestión de diarios en IBM i**

Como parte de la estrategia de copia de seguridad, conviene que se ocupe de los receptores de diario. Es conveniente eliminar los receptores de diario de las bibliotecas de IBM MQ por varias razones:

- Para liberar espacio; esto es válido para todos los receptores de diario

- Para mejorar el rendimiento al arrancar (STRMQM)
- Para mejorar el rendimiento al volver a crear objetos (RCRMQMOBJ)

Antes de suprimir un receptor de diario, asegúrese de que dispone de una copia y de que ya no necesita el receptor de diario.

Los receptores de diario se pueden eliminar de la biblioteca del gestor de colas *después* de desconectarlos de los diarios y de guardarlos, siempre que estén disponibles para restauración si se necesitan en una operación de recuperación.

El concepto de la gestión de diarios se muestra en la [Figura 29 en la página 392](#).

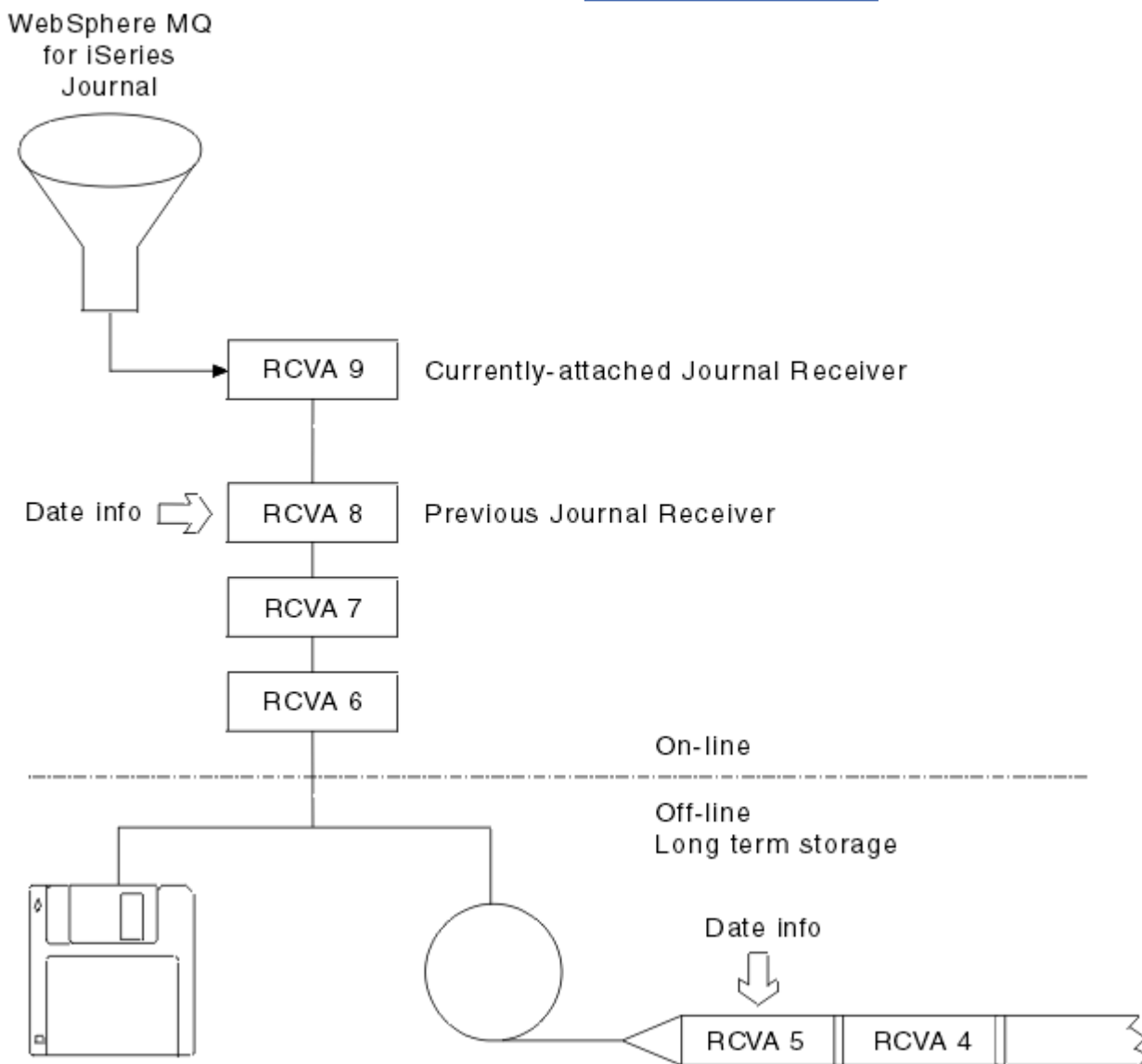


Figura 29. Registro por diario en IBM i

Conviene saber a qué punto anterior de los diarios es probable que tenga que retroceder IBM MQ para poder determinar cuándo se puede eliminar un receptor de diario de la biblioteca del gestor de colas cuya copia de seguridad se haya realizado y cuándo se puede descartar la propia copia de seguridad.

IBM MQ emite dos mensajes a la cola de mensajes del gestor de colas (QMQMMSG en la biblioteca del gestor de colas) para ayudar a determinar ese momento. Estos mensajes se emiten cuando se inicia, cuando cambia un receptor de diario local y se utiliza RCDMQIMG para forzar un punto de comprobación. Los dos mensajes son:



## AMQ7460

Punto de recuperación de arranque. Este mensaje define la fecha y hora de la entrada de arranque a partir de la cual IBM MQ reproduce el diario en el caso de que se produzca un paso de recuperación de arranque. Si el receptor de diario que contiene este registro está disponible en las bibliotecas de IBM MQ, este mensaje también incluye el nombre del receptor de diario que contiene el registro.

## AMQ7462

Entrada más antigua de recuperación desde soporte. Este mensaje define la fecha y hora de la entrada más antigua que se utilizará para volver a crear un objeto a partir de su imagen de soporte.

El receptor de diario identificado es el más antiguo que se necesita. Los receptores de diario de IBM MQ cuyas fechas de creación sean todavía más antiguas ya no se necesitan. Si sólo se visualizan asteriscos, es preciso restaurar las copias de seguridad a partir de la fecha indicada para determinar cuál es el receptor de diario más antiguo.

Cuando se anotan estos mensajes, IBM MQ también graba un objeto de espacio de usuario en la biblioteca del gestor de colas que contiene una sola entrada: el nombre del receptor de diario más antiguo que debe conservarse en el sistema. Este espacio de usuario se denomina AMQJRNINF y los datos se graban en el siguiente formato:

```
JJJJJJJJJJLLLLLLLLLLLLYYYYMMDDHHMMSSmmm
```

donde:

### JJJJJJJJJJ

Es el nombre del receptor más antiguo que IBM MQ todavía necesita.

### LLLLLLLLLL

Es el nombre de la biblioteca de receptores de diario.

### YYYY

Es el año de la entrada de diario más antigua que IBM MQ necesita.

### MM

Es el mes de la entrada de diario más antigua que IBM MQ necesita.

### DD

Es el día de la entrada de diario más antigua que IBM MQ necesita.

### HH

Es la hora de la entrada de diario más antigua que IBM MQ necesita.

### SS

Son los segundos de la entrada de diario más antigua que IBM MQ necesita.

### mmm

Son los milisegundos de la entrada de diario más antigua que IBM MQ necesita.

Una vez que se ha eliminado del sistema el receptor de diario más antiguo, este espacio de usuario contiene asteriscos (\*) para el nombre del receptor de diario.

**Nota:** La realización periódica de RCDMQMIMG OBJ(\*ALL) OBJTYPE(\*ALL) DSPJRNDTA(\*YES) puede ahorrar tiempo de inicio para IBM MQ y reducir el número de receptores de diario locales que necesita salvar y restaurar para la recuperación.

IBM MQ for IBM i no hace referencia a los receptores de diario a menos que esté realizando un paso de recuperación, ya sea para el inicio o para volver a crear un objeto. Si detecta que un diario necesario no está presente, emite el mensaje AMQ7432 a la cola de mensajes del gestor de colas (QMQMMSG) para notificar la fecha y hora de la entrada de diario que necesita para completar el paso de recuperación.

Si esto ocurriera, restaure a partir de la copia de seguridad todos los receptores de diario desconectados después de esa fecha, para permitir que sea satisfactorio el paso de recuperación.

Mantenga el receptor de diario que contiene la entrada de arranque, y todos los receptores de diario posteriores, disponibles en la biblioteca del gestor de colas.

Mantenga el receptor de diario que contiene el `Media Recovery Entry` más antiguo, y cualquier receptor de diario posterior, disponible en todo momento, y presente en la biblioteca del gestor de colas o de la que se haya realizado una copia de seguridad.

Cuando se fuerza un punto de comprobación:

- Si el receptor de diario especificado en AMQ7460 no está avanzado, indica que hay una unidad de trabajo incompleta que debe confirmarse o restituirse.
- Si el receptor de diario especificado en AMQ7462 no está avanzado, indica hay uno o más objetos dañados.

**IBM i**

### **Restauración de un gestor de colas completo (datos y diarios) en IBM i**

Utilice esta información para restaurar uno o más gestores de colas de una copia de seguridad o desde una máquina remota.

Si tiene que recuperar uno o varios gestores de colas de IBM MQ a partir de una copia de seguridad, siga estos pasos:

1. Desactive temporalmente los gestores de colas de IBM MQ.
2. Localice el juego de copias de seguridad más reciente, que consta de la última copia de seguridad completa y de los receptores de diario de los que posteriormente se ha hecho copia de seguridad.
3. Realice una operación `RSTLIB`, a partir de la copia de seguridad completa, para restaurar las bibliotecas de datos de IBM MQ al estado que tenían en el momento de la copia de seguridad completa; para ello, emita estos mandatos:

```
RSTLIB LIB(QMQLIB1) .....  
RSTLIB LIB(QMQLIB2) .....
```

Si un receptor de diario se ha guardado parcialmente en una copia de seguridad de diario y luego se ha guardado completamente en una copia de seguridad posterior, restaure sólo el que se ha guardado completamente. Restaure los diarios de forma individual y por orden cronológico.

4. Realice una operación `RST` para restaurar los directorios IFS de IBM MQ en el sistema de archivos IFS; para ello, emita el siguiente mandato:

```
RST DEV(...) OBJ(('QIBM/UserData/mqm/qmgrs/testqm')) ...
```

5. Inicie el gestor de colas de mensajes. Se reproducen todos los registros de diario grabados desde la copia de seguridad completa y se restauran todos los objetos de IBM MQ al estado coherente que tenían en el momento de la copia de seguridad del diario.

Si desea restaurar un gestor de colas completo en una máquina distinta, utilice el procedimiento siguiente para restaurarlo todo desde la biblioteca del gestor de colas. (TEST es el nombre que se utiliza para el gestor de colas de ejemplo.)

1. `CRTMQM TEST`
2. `DLTLIB LIB(QMTEST)`
3. `RSTLIB SAVLIB(QMTEST) DEV(*SAVF) SAVF(QMGRLIBSAV)`
4. Suprima los siguientes archivos IFS:

```
/QIBM/UserData/mqm/qmgrs/TEST/QMQMCHKPT  
/QIBM/UserData/mqm/qmgrs/TEST/qmanager/QMQMOBJCAT  
/QIBM/UserData/mqm/qmgrs/TEST/qmanager/QMANAGER  
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.AUTH.DATA.QUEUE/q  
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.CHANNEL.INITQ/q  
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.CLUSTER.COMMAND.QUEUE/q  
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.CLUSTER.REPOSITORY.QUEUE/q  
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.CLUSTER.TRANSMIT.QUEUE/q
```

```
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.PENDING.DATA.QUEUE/q
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.ADMIN.COMMAND.QUEUE/q
```

5. STRMQM TEST

6. RCRMQMOBJ OBJ(\*ALL) OBJTYPE(\*ALL) MQMNAME(TEST)

## **IBM i Restauración de los receptores de diario de un determinado gestor de colas en IBM i**

Utilice esta información para conocer los diferentes procedimientos para restaurar receptores de diario.

Si se ha eliminado un receptor que se necesita otra vez para una función de recuperación ulterior, la acción más corriente consiste en restaurar en una biblioteca del gestor de colas un receptor de diario de una copia de seguridad.

Ésta es una tarea simple, en la que los receptores de diario se deben restaurar con el mandato estándar de IBM i RSTOBJ:

```
RSTOBJ OBJ(QMQMDATA/AMQA000005) OBJTYPE(*JRNRVC) .....
```

Puede suceder que se tenga que restaurar una serie de receptores de diario, en lugar de tan solo un receptor. Por ejemplo, supongamos que AMQA000007 es el receptor más antiguo en las bibliotecas de IBM MQ y que se tienen que restaurar los receptores AMQA000005 y AMQA000006.

En este caso, restaure los receptores individualmente en orden cronológico inverso. No siempre es necesario hacerlo así, pero es muy conveniente. En situaciones graves, es posible que necesite utilizar el mandato de IBM i WRKJRNA para asociar los receptores de diario restaurados al diario.

Cuando se restauran diarios, el sistema crea automáticamente un receptor de diario conectado dándole un nuevo nombre en la secuencia de receptores de diario. Sin embargo, el nuevo nombre generado podría no coincidir con el del receptor de diario que se debe restaurar. Para salir al paso de este problema, se necesita intervención manual para crear en secuencia un receptor de diario con un nombre nuevo y un diario nuevo antes de restaurar el receptor de diario.

Por ejemplo, supongamos que surge este problema con respecto al diario guardado AMQAJRN y estos receptores de diario:

- AMQA000000
- AMQA100000
- AMQA200000
- AMQA300000
- AMQA400000
- AMQA500000
- AMQA600000
- AMQA700000
- AMQA800000
- AMQA900000

Al restaurar el diario AMQAJRN en una biblioteca del gestor de colas, el sistema crea automáticamente el receptor de diario AMQA000000. Este receptor generado de forma automática está en conflicto con uno de los receptores de diario existentes (AMQA000000) que desea restaurar y que será imposible hacerlo.

La solución sería:

1. Crear manualmente el siguiente receptor de diario (vea [“Diarios del gestor de colas en IBM i”](#) en la página 386):

```
CRTJRNRVC JRNRVC(QMQRLIB/AMQA900001) THRESHOLD(XXXXX)
```

2. Crear manualmente el diario con el receptor de diario:

```
CRTJRN JRN(QMGRLIB/AMQAJRN) MNGRCV(*SYSTEM) +  
JRNRVC(QMGRLIB/AMQA9000001) MSGQ(QMGRLIB/AMQAJRNMSG)
```

3. Restaurar los receptores de diario locales comprendidos entre AMQA000000 y AMQA900000.

## **IBM i Gestores de colas multiinstancia en IBM i**

Los gestores de colas multiinstancia mejoran la disponibilidad conmutando automáticamente a un servidor en espera si el servidor activo falla. Los servidores activos y en espera son instancias múltiples del mismo gestor de colas; comparten los mismos datos del gestor de colas. Si la instancia activa falla, debe transferir su diario a la que está en espera pues asume que el gestor de colas puede volver a estructurar sus colas.

Configure los sistemas IBM i en los que está ejecutando gestores de colas multiinstancia para que, si la instancia activa de gestor de colas falla, el diario que está utilizando esté disponible para la instancia en espera que la sustituye. Puede diseñar sus propias tareas de configuración y administración para que el diario de la instancia activa esté disponible para la instancia que la sustituye. Si no desea perder mensajes, su diseño debe garantizar que el diario en espera sea consistente con el diario activo cuando se produzca la anomalía. Puede adaptar su diseño a partir de una de las dos configuraciones que se describen con ejemplos en los temas subsiguientes para actualizar la coherencia.

1. Duplique el diario del sistema que está ejecutando la instancia del gestor de colas activa en los sistemas que están ejecutando las instancias en espera.
2. Coloque el diario en una agrupación de almacenamiento auxiliar independiente (Independent Auxiliary Storage Pool - IASP) que se pueda transferir desde el sistema que ejecuta la instancia activa a la instancia en espera.

La primera solución no precisa hardware ni software adicional puesto que utiliza ASP básicas. La segunda solución necesita IASP conmutables que necesitan el soporte de agrupación en clúster de IBM i que está disponible como producto con licencia de IBM i (5761-SS1) opción 41 que se vende por separado.

## **IBM i Fiabilidad y disponibilidad en IBM i**

El objetivo de los gestores de colas multiinstancia es mejorar la disponibilidad de las aplicaciones. Las restricciones tecnológicas y físicas apuntan que el usuario necesita distintas soluciones para poder resarcirse de una recuperación tras desastre, haciendo copias de seguridad de los gestores de colas y garantizando un funcionamiento continuo.

Al configurar la fiabilidad y la disponibilidad debe evaluar un gran número de factores que podrían resumirse en cuatro puntos característicos:

### **Recuperación tras desastre**

Se ha optimizado para la recuperación después de un desastre importante que pueda destruir todos los activos locales.

La recuperación tras desastre en IBM i a menudo se basa en la duplicación geográfica de IASP.

### **Reserva**

Se ha optimizado para poder efectuar la recuperación tras una anomalía localizada, que habitualmente se trata de un error humano o de un problema técnico imprevisto.

IBM MQ proporciona gestores de colas de copia de seguridad para hacer copia de seguridad de los gestores de colas de forma periódica. También puede utilizar la réplica asíncrona de diarios del gestor de colas para mejorar la efectividad de las copias de seguridad.

### **Disponibilidad**

Se ha optimizado para restaurar rápidamente operaciones de forma que adquieran la apariencia de un servicio casi ininterrumpido pues pueden anticiparse a anomalías técnicas previsibles tales como un error de disco o de servidor.

Normalmente, la recuperación se mide en minutos, pero la detección a veces se prolonga más que el propio proceso de recuperación. Un gestor de colas multiinstancia le ayuda a configurar la *disponibilidad*.

### **Operación continua**

Se ha optimizado para suministrar un servicio ininterrumpido.

Las soluciones de operación continua deben solucionar el problema de detección, lo que casi siempre implica someter el mismo trabajo a través de más de un sistema y, o bien utilizar el primer resultado o, si la exactitud de los datos constituye el objetivo principal, comparar como mínimo dos resultados.

Un gestor de colas multiinstancia le ayuda a configurar la *disponibilidad*. Sólo hay una instancia del gestor de colas activa a la vez. El cambiar a una instancia en espera se puede prolongar desde un poco más de diez segundos hasta unos quince minutos o más, en función de cómo se haya configurado, cargado y ajustado el sistema.

Un gestor de colas de varias instancias puede dar la apariencia de un servicio casi ininterrumpido si se utiliza con IBM MQ MQI clients reconectables, que son capaces de continuar el proceso sin que el programa de aplicación sepa realmente que se ha producido una interrupción del gestor de colas; consulte el tema [Reconexión de cliente automatizada](#).

**IBM i**

### **Componentes de una solución de alta disponibilidad en IBM i**

Construya una solución de alta disponibilidad utilizando gestores de colas gestor de colas multiinstancia proporcionando un almacenamiento por red robusto para los datos del gestor de colas, réplica de diarios o almacenamiento IASP robusto para diarios de gestor de colas y utilizando clientes que se puedan volver a conectar (de aplicaciones configuradas como servicios de gestor de colas reiniciables).

Al detectar una anomalía del gestor de colas, un gestor de colas multiinstancia reacciona reanudando el inicio de otra instancia de gestor de colas en otro servidor. Para completar dicho inicio, la instancia debe acceder a los datos del gestor de colas compartidos en almacenamiento por red y a su copia del diario de gestor de colas local.

Para crear una solución de alta disponibilidad, debe gestionar la disponibilidad de los datos del gestor de colas, la vigencia del diario del gestor de colas local y, o bien crear aplicaciones cliente con capacidad de reconexión, o bien desplegar las aplicaciones como servicios de gestor de colas para que se reinicien automáticamente cuando se reanude el gestor de colas. La reconexión automática de cliente no está soportada en IBM MQ classes for Java.

### **Datos del gestor de colas**

Coloque los datos del gestor de colas en almacenamiento por red que se comparta, altamente disponible y fiable, posiblemente utilizando discos RAID de nivel 1 o posteriores. El sistema de archivos debe cumplir con los requisitos de un sistema de archivos compartidos para gestores de colas multiinstancia; con objeto de obtener más información acerca de los requisitos de los sistemas de archivos compartidos, consulte [Requisitos de los sistemas de archivos compartidos](#). Network File System 4 (NFS4) es un protocolo que cumple con estos requisitos.

### **Diarios de gestor de colas**

También debe configurar los diarios de IBM i utilizados por las instancias del gestor de colas de forma que la instancia en espera pueda restaurar sus datos del gestor de colas a un estado coherente. En caso de interrupción del servicio, significa que deberá restaurar los diarios a su estado cuando la instancia activa falle. A diferencia de las soluciones de copia de seguridad o de recuperación ante siniestro, la restauración de diarios en un punto de comprobación anterior no es suficiente.

No puede compartir físicamente diarios entre varios sistemas IBM i en almacenamiento por red. Para restaurar diarios del gestor de colas a un estado coherente en el momento en que se produce la anomalía, o bien deberá transferir el diario físico que era local para la instancia del gestor de colas activa en el momento de la anomalía a una instancia nueva que se haya activado, o bien deberá actualizar duplicados del diario en instancias en espera en ejecución. El diario duplicado es una réplica del diario remoto que se ha mantenido exactamente en sincronización con el diario local que pertenece a la instancia fallida.

Existen tres configuraciones que constituyen puntos de inicio para designar cómo se gestionan los diarios para un gestor de colas multiinstancia,

1. la utilización de la réplica de diarios sincronizada (duplicación de diarios) desde la ASP de la instancia activa a las ASP de instancias en espera.
2. al transferir una IASP, ha efectuado una configuración para mantener el diario del gestor de colas desde la instancia activa a la instancia en espera que va a adoptar el papel de instancia activa.
3. la utilización duplicados de IASP secundarios sincronizados.

Consulte las opciones de [ASP](#), para obtener más información sobre cómo poner los datos del gestor de colas en un iASP, en el mandato CRTMQM de IBM MQ IBM i.

Consulte también [Alta disponibilidad](#) en la información de IBM i en IBM Documentation.

## Aplicaciones

Para crear un cliente para que se reconecte automáticamente al gestor de colas cuando el gestor de colas en espera se reanuda, conecte su aplicación al gestor de colas utilizando MQCONNX y especifique MQCNO\_RECONNECT\_Q\_MGR en el campo **MQCNO** Opciones. Consulte [Programas de ejemplo de alta disponibilidad](#) para obtener tres programas de muestra utilizando clientes con capacidad de reconexión y [Recuperación de la aplicación](#) para obtener información sobre cómo diseñar aplicaciones cliente para la recuperación.

### Creación de una unidad compartida de datos de un gestor de colas utilizando NetServer en IBM i

Cree una unidad compartida de red en un servidor IBM i para almacenar los datos de gestor de colas. Configure las conexiones de dos servidores, en los que se vayan a albergar la instancias del gestor de colas, para poder acceder a la unidad compartida de red.

## Antes de empezar

- Para esta tarea se necesitan tres servidores IBM i. La unidad compartida de red se define en uno de los servidores, GAMMA. Los otros dos servidores, ALPHA y BETA, se conectarán a GAMMA.
- Instale IBM MQ en los tres servidores.
- Instale System i Navigator; consulte [System i Navigator](#).

## Acerca de esta tarea

- Cree el directorio del gestor de colas en GAMMA, y establezca la propiedad y los permisos correctos para los perfiles de usuario QMQM y QMQMADM. El directorio y los permisos se crean fácilmente instalando IBM MQ en GAMMA.
- Utilice System i Navigator para crear una unidad compartida para el directorio de datos del gestor de colas en GAMMA.
- Cree directorios en ALPHA y BETA que señalan al compartimiento.

## Procedimiento

1. En GAMMA, cree el directorio en el que albergar los datos de gestor de colas con el perfil de usuario QMQM como propietario, y QMQMADM como el grupo primario.

### Consejo:

Una forma rápida y fiable de crear el directorio con los permisos correctos es instalar IBM MQ en GAMMA.

Posteriormente, si no desea ejecutar IBM MQ en GAMMA, desinstale IBM MQ. Tras la desinstalación, el directorio /QIBM/UserData/mqm/qmgrs permanece en GAMMA con el perfil de usuario QMQM como propietario, y QMQMADM como grupo primario.

La tarea utiliza el directorio /QIBM/UserData/mqm/qmgrs en GAMMA para el compartimiento.

2. Inicie el asistente System i Navigator **Añadir conexión** y conéctese al sistema GAMMA.
  - a) Efectúe una doble pulsación en el icono **System i Navigator** del escritorio de Windows.
  - b) Pulse **Sí** para crear una conexión.
  - c) Siga las instrucciones del asistente **Añadir conexión** y cree una conexión del sistema IBM i a GAMMA.

La conexión a GAMMA se añade a **Mis conexiones**.

3. Añada un compartimiento de archivos nuevo en GAMMA.
  - a) En la ventana **System i Navigator**, pulse la carpeta File Shares en My Connections/GAMMA/File Systems.
  - b) En la ventana **Mis tareas**, pulse **Gestionar particiones de IBM i NetServer**.

En el escritorio se abre una nueva ventana, **IBM i NetServer - GAMMA**, en la que se muestran los objetos compartidos.
  - c) Pulse con el botón derecho del ratón en la carpeta Shared Objects > **Archivo** > **Nuevo** > **Archivo**.

Se abre una nueva ventana **Compartición de archivos de IBM i NetServer - GAMMA**.
  - d) Asigne un nombre a la unidad compartida, por ejemplo, WMQ.
  - e) Establezca el control de acceso en Read/Write.
  - f) Seleccione **Nombre de vía de acceso** examinando el directorio /QIBM/UserData/mqm/qmgrs que ha creado anteriormente y pulse **Aceptar**.

La ventana **Compartimiento de archivos de IBM i NetServer-GAMMA** se cierra y WMQ se lista en la ventana de objetos compartidos.

4. Pulse el botón derecho del ratón **WMQ**, en la ventana de objetos compartidos. Pulse **Archivo** > **Permisos**.

Se abre una ventana, **Permisos Qmgrs - GAMMA**, para el objeto /QIBM/UserData/mqm/qmgrs.

- a) Compruebe los permisos siguientes para QMQM, si todavía no se han establecido:

- Read
- Write
- Execute
- Management
- Existence
- Alter
- Reference

- b) Compruebe los permisos siguientes para QMQMADM, si todavía no se han establecido:

- Read
- Write
- Execute
- Reference

- c) Añada otros perfiles de usuario que desee otorgar a /QIBM/UserData/mqm/qmgrs.

Por ejemplo, puede otorgar al perfil de usuario predeterminado (Public) los permisos Read y Execute para /QIBM/UserData/mqm/qmgrs.

5. Compruebe que todos los perfiles de usuario a los que se ha otorgado acceso a /QIBM/UserData/mqm/qmgrs en GAMMA tengan la misma contraseña que en los servidores que acceden a GAMMA.

En concreto, asegúrese de que los perfiles de usuario QMQM en otros servidores, que vayan a acceder a la unidad compartida, tengan la misma contraseña que el perfil de usuario QMQM en GAMMA.

**Consejo:** Pulse la carpeta My Connections/GAMMA/Users and Groups en System i Navigator para establecer las contraseñas. También puede utilizar los mandatos **CHFUSRPRF** y **CHGPWD**.

## Resultados

Compruebe que pueda acceder a GAMMA, desde otros servidores, mediante la unidad compartida. Si está realizando las demás tareas, compruebe que puede acceder a GAMMA desde ALPHA y BETA utilizando la vía de acceso /QNTC/GAMMA/WMQ. Si el directorio /QNTC/GAMMA no existe en ALPHA o BETA, debe crear el directorio. En función del dominio de NetServer, deberá realizar una IPL en ALPHA o en BETA, antes de crear el directorio.

```
CRTDIR DIR('/QNTC/GAMMA')
```

Cuando haya comprobado que tiene acceso a /QNTC/GAMMA/WMQ desde ALPHA o BETA, emitiendo el mandato, `CRTMQM MQMNAME('QM1') MQMDIRP('/QNTC/GAMMA/WMQ')` crea /QIBM/UseData/mqm/qmgrs/QM1 en GAMMA.

## Qué hacer a continuación

Cree un gestor de colas multiinstancia siguiendo los pasos que figuran en cualquiera de las tareas, [“Creación de un gestor de colas multiinstancia mediante la duplicación de diarios y NetServer en IBM i”](#) en la página 411 o [“Conversión de un gestor de colas de instancia única en un gestor de colas multiinstancia usando NetServer y duplicación de diarios en IBM i”](#) en la página 415.

### **Rendimiento de una migración tras error en IBM i**

El tiempo que se tarda en detectar si una instancia de gestor de colas ha fallado y, a continuación, para reanudar el procesamiento en modo de espera puede variar entre decenas de segundos hasta quince minutos o más, dependiendo de la configuración. El rendimiento debe ser un importante punto a tener en cuenta al diseñar y probar una solución de alta disponibilidad.

Existen ventajas y desventajas que se deben sopesar al decidir si configurar un gestor de colas multiinstancia para que utilice la réplica de diario o si utilizar una IASP. La duplicación necesita que el gestor de colas grabe síncronamente en un diario remoto. Desde el punto de vista del hardware, esta necesidad no afecta al rendimiento, pero desde el punto de vista del software se implica una longitud de vía de acceso superior al escribir en un diario remoto que si sólo se hace en un diario local, por lo que puede deducirse que el rendimiento se verá de alguna manera reducido al ejecutar un gestor de colas. No obstante, cuando el gestor de colas en espera realiza la sustitución, el retardo al sincronizar su diario local desde el diario remoto actualizado por la instancia activa antes de su fallo, normalmente resulta pequeño en comparación con el tiempo que IBM i necesita para detectar y transferir la IASP al servidor que ejecuta la instancia en espera del gestor de colas. Los tiempos de transferencia de la IASP pueden alcanzar de diez a quince minutos, en lugar de completarse en segundos. El tiempo de transferencia de la IASP depende del número de objetos que deban estar *disponibles* cuando la IASP se transfiere al sistema en espera y del tamaño de las vías de acceso o los índices, puesto que deben fusionarse.

Cuando el gestor de colas en espera realiza la sustitución, el retardo al sincronizar su diario local desde el diario remoto actualizado por la instancia activa antes de su fallo, normalmente resulta pequeño en comparación con el tiempo que IBM i necesita para detectar y transferir la ASP independiente al servidor que ejecuta la instancia en espera del gestor de colas. Los tiempos de transferencia de ASP independientes pueden alcanzar de diez a quince minutos, en lugar de completarse en segundos. El tiempo de transferencia de la ASP independiente depende del número de objetos que deban estar *disponibles* cuando la ASP independiente se transfiera al sistema en espera y del tamaño de las vías de acceso o los índices, puesto que deben fusionarse.

No obstante, la transferencia del diario no es el único factor que influye en el tiempo que la instancia en espera necesita para reanudarse de forma completa. También es necesario considerar el tiempo que tarda el sistema de archivos de red en liberar el bloqueo de los datos del gestor de colas que señala a la instancia en espera para tratar de continuar con su puesta en marcha, y también el tiempo necesario para recuperar las colas del diario para que la instancia pueda empezar a procesar los mensajes de nuevo.



Deben añadirse todos estos otros orígenes de retardo al tiempo que se tarda en iniciar una instancia de espera. El tiempo total para cambiar consta de los siguientes componentes,

#### **Tiempo de detección de anomalía**

El tiempo que NFS necesita para liberar el bloqueo de los datos del gestor de colas y el que necesita la instancia en espera para continuar su proceso de inicio.

#### **Tiempo de transferencia**

En el caso de un clúster de alta disponibilidad (HA), el tiempo que IBM i tarda en transferir la IASP desde el sistema que hospeda la instancia activa a la instancia en espera, y en el caso de réplica de diario, el tiempo que tarda para actualizar el diario local en espera con los datos de la réplica remota.

#### **Tiempo de inicio**

El tiempo que tarda la nueva instancia del gestor de colas activa en reconstruir sus colas desde el último punto de comprobación en su diario restaurado y en reanudar el procesamiento de mensajes.

#### **Nota:**

Si la instancia en espera que ha que ha realizado la sustitución está configurada para replicarse de forma síncrona con la instancia anteriormente activa, el inicio podría retardarse durante mucho tiempo. Es posible que la instancia nueva activada no sea capaz de replicarse con su diario remoto, si el diario remoto está en el servidor que albergaba la instancia activa anteriormente y el servidor ha fallado.

El valor predeterminado de tiempo de espera para una respuesta síncrona es de un minuto. Puede configurar el retardo máximo antes de que la réplica caduque. O bien, puede configurar instancias en espera para iniciarse utilizando la réplica asíncrona en la instancia activa fallida. Más adelante, conmutará a la réplica síncrona, cuando la instancia fallida se esté ejecutando de nuevo en espera. La misma consideración se aplica al uso de duplicaciones ASP independientes síncronas.

Puede hacer mediciones de línea base separadas para estos componentes que le ayudarán a evaluar el tiempo global que le ha conducido hasta la sustitución por anomalía y a incidir en la decisión sobre el método de configuración que debe utilizar. Para tomar la mejor decisión en cuanto a la configuración, también debe tener en cuenta cómo harán la sustitución por anomalía las otras aplicaciones del mismo servidor y si hay otros procesos de copias de seguridad o de recuperación tras desastre que ya utilicen la IASP.

Los tiempos de transferencia de la IASP se pueden reducir ajustando la configuración del clúster:

1. Los perfiles de usuario de todos los sistemas del clúster deberían tener el mismo UID y GID para que no sea necesario que los procesos habilitados deban cambiar los UID y los GID.
2. Puede minimizar el número de objetos de base de datos del sistema y de agrupaciones de discos de usuario, ya que éstos deben ser fusionarse para crear la tabla de referencias cruzadas para el grupo de agrupación de discos.
3. Encontrará más consejos para mejorar el rendimiento en el IBM Redbook, *Implementing PowerHA for IBM i, SG24-7405*.

Una configuración utilizando ASP básicas, la duplicación de diarios y una pequeña configuración debería cambiar en nada más y nada menos que en decenas de segundos.

### **IBM i *Visión general de la combinación de las prestaciones de agrupación en clúster de IBM i con la agrupación en clúster de IBM MQ***

La ejecución de IBM MQ en IBM i y la utilización de las prestaciones de agrupación en clúster de IBM i puede proporcionar una solución de alta disponibilidad más completa que la utilización de la agrupación en clúster de IBM MQ únicamente.

Para tener esta prestación, debe configurar:

1. Los clústeres en la máquina de IBM i; consulte [“Clústeres de IBM i” en la página 402](#)
2. Una agrupación de almacenamiento auxiliar independiente (IASP), a la que mueve el gestor de colas; consulte [“Agrupaciones de almacenamiento auxiliar independientes \(IASP\)” en la página 402](#)

3. Un grupo de recursos de clúster (CRG); consulte [“Grupos de recursos de clúster de dispositivos”](#) en la [página 402](#), en el que define:

- El dominio de recuperación
- La IASP
- El programa de salida; consulte [“Programa de salida de CRG de dispositivo”](#) en la [página 403](#)

## Clústeres de IBM i

Un clúster de IBM i es una colección de instancias, es decir, sistemas o particiones de IBM i que están enlazados lógicamente.

La finalidad de esta agrupación es permitir la copia de seguridad de cada una de las instancias, eliminando un único punto de anomalía y aumentando la flexibilidad de los datos y las aplicaciones. Una vez que se ha creado un clúster, pueden gestionarse los distintos tipos de grupo de recursos de clúster (CRG) para gestionar aplicaciones, datos y dispositivos en el clúster.

Para obtener más información, consulte [Creación de un clúster](#) y el mandato [Crear clúster \(CRTCLU\)](#).

## Agrupaciones de almacenamiento auxiliar independientes (IASP)

Una IASP es un tipo de ASP de usuario que se utiliza como una extensión de un almacenamiento de un solo nivel. Es un elemento de almacenamiento que, debido a su independencia del almacenamiento del sistema, se puede manipular fácilmente sin tener que realizar una IPL del sistema.

Una IASP se puede conmutar fácilmente a otra instancia de sistema operativo o replicar en una IASP de destino en otra instancia de sistema operativo. Se pueden utilizar los métodos para conmutar una IASP entre instancias:

- El primer método requiere que todos los sistemas del clúster, y la torre de discos conmutables que contiene la IASP, estén conectados mediante un bucle HSL (High Speed Link).
- El segundo método requiere que las instancias del sistema operativo sean particiones en el mismo sistema de IBM i donde los procesadores de entrada/salida (IOP) se pueden conmutar entre particiones. No se requiere ningún hardware especial para poder replicar una IASP. La réplica se realiza mediante TCP/IP a través de la red.

Para obtener más información, consulte el mandato [Configurar dispositivo ASP \(CFGDEVASP\)](#).

## Grupos de recursos de clúster de dispositivos

Hay distintos tipos de grupos de recursos de clúster (CRG). Para obtener más información sobre los distintos tipos de CRG disponibles, consulte [Grupo de recursos de clúster](#).

Este tema se concentra en un CRG de dispositivo. Un CRG de dispositivo:

- Describe y gestiona los recursos de dispositivo como por ejemplo las agrupaciones de almacenamiento auxiliar independientes (IASP).
- Define el dominio de recuperación de los nodos de clúster
- Asigna un dispositivo, y
- Asigna el programa de salida que gestionará los sucesos del clúster.

El dominio de recuperación indica qué nodo del clúster se considerará el nodo primario. Los demás nodos se consideran copias de seguridad. Los nodos de copia de seguridad también están ordenados en el dominio de recuperación, y se especifica qué nodo es la primera copia de seguridad, la segunda copia de seguridad, etc., según el número de nodos que haya en el dominio de recuperación.

En caso de una anomalía de nodo primario, el programa de salida se ejecuta en todos los nodos del dominio de recuperación. A continuación, el programa de salida en ejecución en la primera copia de seguridad puede realizar las inicializaciones necesarias para convertir este nodo en el nuevo nodo primario.

Para obtener más información, consulte [Creación de CRG de dispositivo](#) y el mandato [Crear grupo de recursos de clúster \(CRTCRG\)](#).

## Programa de salida de CRG de dispositivo

El servicio de recursos del clúster del sistema operativo llama a un programa de salida de CRG de dispositivo cuando se produce un suceso en uno de los nodos que define el dominio de recuperación; por ejemplo, un suceso de sustitución por anomalía o conmutación.

Un suceso de sustitución por anomalía se produce cuando el nodo primario del clúster falla y los CRG se conmutan con todos los recursos que gestionan, y un suceso de conmutación se produce cuando un CRG específico se conmuta manualmente desde el nodo primario al nodo de copia de seguridad.

En cualquiera de los dos casos, el programa de salida es el responsable de inicializar e iniciar todos los programas que estaban en ejecución en el nodo primario anterior, lo que convierte el primer nodo de copia de seguridad en el nuevo nodo primario.

Por ejemplo, con IBM MQ, el programa de salida debería ser el responsable de iniciar el subsistema de IBM MQ (QMQM) y los gestores de colas. Los gestores de colas se deberían configurar para iniciar automáticamente los escuchas y servicios, como por ejemplo los supervisores.

**V 9.1.0** Está disponible un programa de salida de ejemplo, AMQSCRG4, en IBM i desde IBM MQ 9.1.

## Configuración de IASP conmutable

Se puede configurar IBM MQ para aprovechar las prestaciones de agrupación en clúster de IBM i. Para ello:

1. Cree un clúster de IBM i entre los sistemas de centro de datos
2. Mueva el gestor de colas a una IASP.

“Mover, o eliminar, un gestor de colas a, o desde, una agrupación de almacenamiento auxiliar independiente” en la página 404 contiene código de ejemplo para ayudarle a realizar esta operación.

3. Debe crear un CRG definiendo el dominio de recuperación, la IASP y el programa de salida.

“Configuración de un grupo de recursos de clúster de dispositivo” en la página 403 contiene código de ejemplo para ayudarle a realizar esta operación.

### Conceptos relacionados

“ASPs independientes y alta disponibilidad” en la página 424

Las ASP independientes permiten mover aplicaciones y datos entre servidores. La flexibilidad de las ASP independientes las ha convertido en la base de algunas soluciones de alta disponibilidad de IBM i. Cuando se plantee la posibilidad de utilizar una ASP o una ASP independiente para el diario de gestor de colas, debería tener en cuenta otra configuración de alta disponibilidad basada en ASP independientes.

**IBM i** *Configuración de un grupo de recursos de clúster de dispositivo*

Programa de ejemplo para configurar un grupo de recursos de clúster de dispositivo (CRG).

## Acerca de esta tarea

En el ejemplo siguiente, tenga en cuenta que:

- [NOMBRE SITIO PRIMARIO] y [NOMBRE SITIO COPIA DE SEGURIDAD] podrían ser dos series distintas cualesquiera de ocho caracteres o menos.
- [IP PRIMARIA] e [IP DE COPIA DE SEGURIDAD] son las IP que se utilizarán para la duplicación.

## Procedimiento

1. Identifique el nombre del clúster.
2. Identifique el nombre de programa de salida de CRG y la biblioteca.

3. Determine el nombre del nodo primario y de los nodos de copia de seguridad que definirá este CRG.
4. Identifique la IASP que gestionará este CRG, y asegúrese de que se ha creado en el nodo primario.
5. Cree una descripción de dispositivo en los nodos de copia de seguridad mediante este mandato:

```
CRTDEVASP DEVD([IASP NAME]) RSRNAME([IASP NAME])
```

6. Añada la dirección IP de toma de control a todos los nodos mediante el mandato:

```
ADDTCPICF INTNETADR(' [TAKEOVER IP]') LIND([LINE DESC])
SUBNETMASK(' [SUBNET MASK]') AUTOSTART(*NO)
```

7. Inicie la dirección IP de toma de control solo en el nodo primario mediante el mandato:


```
STRTCPICF INTNETADR(' [TAKEOVER IP]')
```

8. Opcional: Si la IASP es conmutable, invoque este mandato:

```
CRTCRG CLUSTER([CLUSTER NAME]) CRG( [CRG NAME]) CRGTYPE(*DEV) EXITPGM([EXIT LIB]/[EXIT
NAME])
USRPRF([EXIT PROFILE]) RCYDMN(( [PRIMARY NODE] *PRIMARY) ([BACKUP NAME] *BACKUP))
EXITPGMFMF(EXTP0200) CFGOBJ([IASP NAME] *DEVD *ONLINE '[TAKEOVER IP]')
```

9. Opcional: Si se va a duplicar la IASP, invoque este mandato:

```
CRTCRG CLUSTER([CLUSTER NAME]) CRG([CRG NAME]) CRGTYPE(*DEV) EXITPGM([EXIT LIB]/[EXIT NAME])
USRPRF([EXIT PROFILE]) RCYDMN(( [PRIMARY NODE] *PRIMARY *LAST [PRIMARY SITE NAME] (' [PRIMARY
IP] '))
[BACKUP NAME] *BACKUP *LAST [BACKUP SITE NAME] (' [BACKUP IP] ')) EXITPGMFMF(EXTP0200)
CFGOBJ([IASP NAME] *DEVD *ONLINE '[TAKEOVER IP]')
```

 *Mover, o eliminar, un gestor de colas a, o desde, una agrupación de almacenamiento auxiliar independiente*

Un programa de ejemplo para mover un gestor de colas a una agrupación de almacenamiento auxiliar independiente (IASP) y mandatos para eliminar un gestor de colas de una IASP.

## Acerca de esta tarea

En el ejemplo siguiente, tenga en cuenta que:

- [NOMBRE GESTOR] es el nombre del gestor de colas.
- [NOMBRE IASP] es el nombre de la IASP.
- [BIBLIOTECA GESTOR] es el nombre de la biblioteca del gestor de colas.
- [DIRECTORIO GESTOR] es el nombre del directorio del gestor de colas.

## Procedimiento

1. Identifique el nodo primario y los nodos de copia de seguridad.
2. Realice el procedimiento siguiente en el nodo primario:
  - a) Asegúrese de que el gestor de colas ha finalizado.
  - b) Asegúrese de que la IASP esté establecida en vary on mediante el mandato

```
VRYCFG CFGOBJ([IASP NAME]) CFGTYPE(*DEV) STATUS(*ON)
```

- c) Cree el directorio de gestores de colas bajo la IASP.  
Habrá un directorio bajo la raíz con el nombre de la IASP, que es:

```
QSH CMD('mkdir -p /[IASP_NAME]/QIBM/UserData/mqm/qmgrs/')
```

- d) Mueva los objetos de IFS del gestor al directorio de gestores de colas que acaba de crear bajo la IASP mediante el mandato siguiente:

```
QSH CMD('mv /QIBM/UserData/mqm/qmgrs/[MANAGER NAME]
/[IASP NAME]/QIBM/UserData/mqm/qmgrs')
```

- e) Cree un archivo de salvar temporal denominado MGRLIB mediante el mandato:

```
CRTSAVF QGPL/MGRLIB
```

- f) Guarde la biblioteca de gestores de colas en el archivo de salvar MGRLIB, mediante el mandato siguiente:

```
SAVLIB LIB([MANGER LIBRARY]) DEV(*SAVF) SAVF(QGPL/MGRLIB)
```

- g) Suprima la biblioteca de gestores de colas mediante el mandato siguiente, e ignore todos los mensajes de consulta:

```
DLTLIB [MANAGER LIBRARY]
```

- h) Restaure la biblioteca de gestores de colas en la IASP mediante el mandato siguiente:

```
RSTLIB SAVLIB([MANAGER LIBRARY]) DEV(*SAVF) SAVF(QGPL/MGRLIB)
RSTASPDEV([IASP NAME])
```

- i) Suprima el archivo de salvar temporal mediante el mandato siguiente:

```
DLTF FILE(QGPL/MGRLIB)
```

- j) Cree un enlace simbólico a los objetos de IFS del gestor de colas bajo la IASP, mediante el mandato siguiente:

```
ADDLNK OBJ('/[IASP NAME]/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')
NEWLNK('/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')
```

- k) Conecte a la IASP mediante el mandato siguiente:

```
SETASPGRP [IASP NAME]
```

- l) Inicie el gestor de colas mediante el mandato:

```
STRMQM [MANAGER NAME]
```

### 3. Realice el procedimiento siguiente en el nodo o nodos de copia de seguridad:

- a) Cree un directorio de gestores de colas temporal mediante el mandato siguiente:

```
QSH CMD('mkdir -p /[IASP NAME]/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')
```

- b) Cree un enlace simbólico al directorio temporal de gestores de colas mediante el mandato:

```
ADDLNK OBJ('/[IASP NAME]/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')
NEWLNK('/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')
```

- c) Suprima el directorio temporal mediante el mandato siguiente:

```
QSH CMD('rm -r /[IASP NAME]')
```

- d) Añada lo siguiente al final del archivo /QIBM/UserData/mqm/mqs.ini:

```
QueueManager:
Name=[MANAGER NAME]
Prefix=/QIBM/UserData/mqm
Library=[MANAGER LIBRARY]
Directory=[MANAGER DIRECTORY]
```

### 4. Para eliminar un gestor de colas de una IASP, emita los mandatos siguientes:

- a) VRYCFG CFGOBJ([NOMBRE IASP]) CFGTYPE(\*DEV) STATUS(\*ON)
- b) SETASPGRP [NOMBRE IASP]
- c) ENDMQM [NOMBRE GESTOR]
- d) DLTMQM [NOMBRE GESTOR]

**IBM i**

### **Configuración de un diario duplicado para ASP en IBM i**

Configure un gestor de colas de instancias múltiples robusto utilizando una réplica síncrona entre diarios duplicados.

Una configuración de gestor de colas duplicadas utiliza diarios que se crean en agrupaciones de almacenamiento auxiliar (ASP) básicas o independientes.

En IBM i, los datos del gestor de colas se escriben en diarios y en un sistema de archivos. Los diarios contienen la copia maestra de los datos del gestor de colas. Los diarios se comparten entre sistemas mediante la replicación de diarios síncrona o asíncrona. Para reiniciar una instancia de gestor de colas se necesita una mezcla de diarios locales y remotos. El reinicio del gestor de colas lee registros de diario de la mezcla de diarios locales y remotos en el servidor y los datos del gestor de colas en el sistema de archivos de red compartidos. Los datos del sistema de archivos aceleran el reinicio del gestor de colas. Los puntos de comprobación se almacenan en el sistema de archivos, marcando puntos de sincronización entre el sistema de archivos y los diarios. Los registros de diario almacenados antes del punto de comprobación no son necesarios para reinicios típicos del gestor de colas. Sin embargo, los datos del sistema de archivos pueden no estar actualizados y los registros de diario después del punto de comprobación se utilizan para completar el reinicio del gestor de colas. Los datos de los diarios adjuntos a la instancia se mantienen actualizados para que el reinicio se pueda completar satisfactoriamente.

Pero incluso los registros de diario pueden no estar actualizados si el diario remoto del servidor en espera se estaba replicando asíncronamente y la anomalía se ha producido antes de la sincronización. En el caso en que decida reiniciar un gestor de colas mediante un diario remoto no sincronizado, la instancia del gestor de colas en espera puede volver a procesar mensajes suprimidos antes de que la instancia activa haya fallado o no procesar mensajes recibidos antes de que la instancia activa haya fallado.

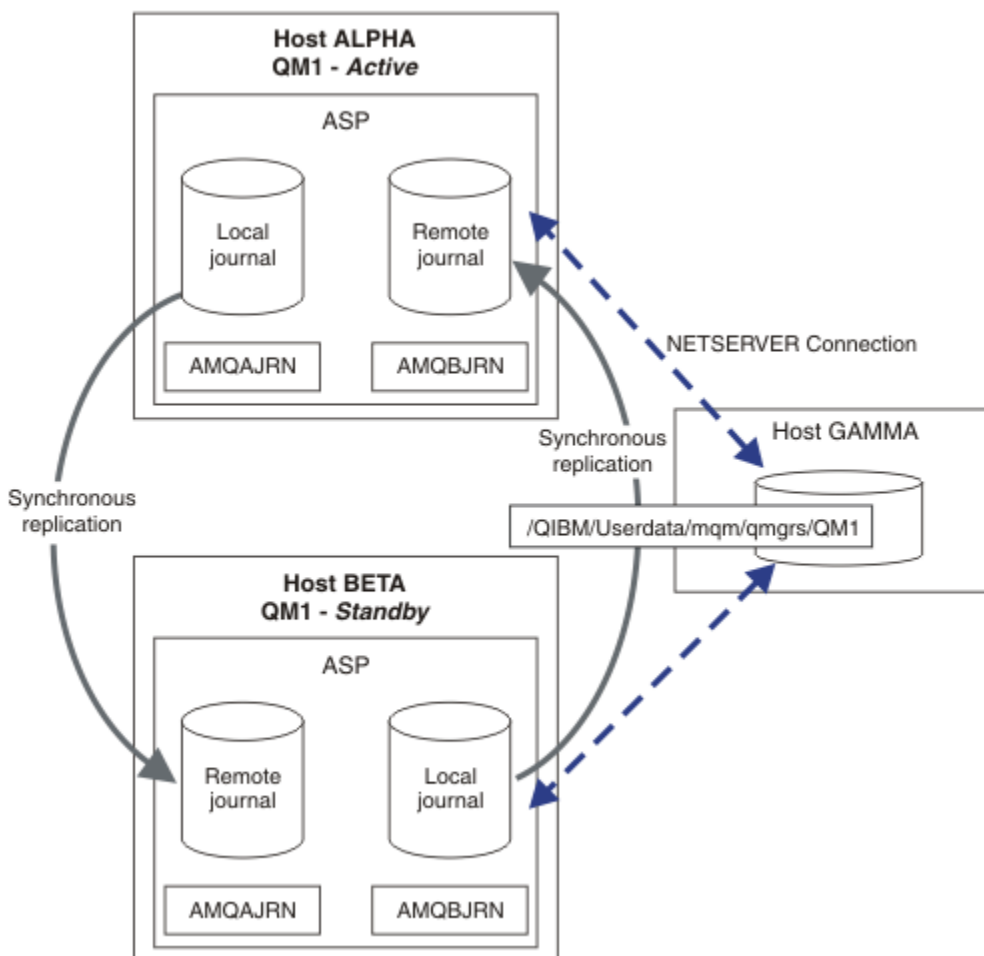
Otra posibilidad aunque remota, es que el sistema de archivos contenga el registro de punto de comprobación más reciente y un diario remoto no sincronizado en espera no lo contenga. En este caso, el gestor de colas no se reinicia automáticamente. Puede esperar hasta que el diario remoto esté sincronizado o iniciar el gestor de colas en espera en frío desde el sistema de archivos. Incluso aunque en este caso el sistema de archivos contenga un punto de comprobación más reciente de los datos del gestor de colas que el diario remoto, no contendrá todos los mensajes procesados antes de la anomalía de la instancia activa. Es posible que algunos mensajes se vuelvan a procesar y que otros no se procesen después de un reinicio en frío que no está sincronizado con los diarios.

Con un gestor de colas multiinstancia, el sistema de archivos también se utiliza para controlar qué instancia de un gestor de colas está activa y cuál está en espera. La instancia activa adquiere un bloqueo para los datos del gestor de colas. La que está en espera, espera para adquirir el bloqueo y cuando lo hace, se convierte en la instancia activa. La instancia activa libera el bloqueo si finaliza normalmente. El sistema de archivos libera el bloqueo si detecta que la instancia activa ha fallado o no puede acceder al sistema de archivos. El sistema de archivos debe cumplir los requisitos para detectar la anomalía; consulte el apartado [Requisitos para sistemas de archivos compartidos](#).

La arquitectura de gestores de colas multiinstancia en IBM i proporciona un reinicio automático después de la anomalía del servidor o el gestor de colas. También soporta la restauración de los datos del gestor de colas después de la anomalía del sistema de archivos en el que está almacenado el gestor de colas.

En la [Figura 30 en la página 407](#), si ALPHA falla, puede reiniciar manualmente QM1 en BETA, mediante el diario duplicado. Al añadir la posibilidad del gestor de colas multiinstancia a QM1, la instancia en espera de QM1 se reanuda automáticamente en BETA si la instancia activa en ALPHA falla. QM1 también se puede reanudar automáticamente si es el servidor ALPHA el que falla, no solo la instancia activa de QM1. Una vez que BETA se convierte en el host de la instancia del gestor de colas activa, la instancia en espera se puede iniciar en ALPHA.

La [Figura 30 en la página 407](#) muestra una configuración que duplica diarios entre dos instancias de un gestor de colas mediante NetServer para almacenar datos de gestor de colas. Es posible que desee expandir el patrón para incluir más diarios y, por consiguiente, más instancias. Siga las reglas de denominación de diarios que se explican en el tema [“Diarios del gestor de colas en IBM i”](#) en la [página 386](#). Actualmente, el número de instancias en ejecución del gestor de colas está limitado a dos, uno está activo y el otro está en espera.



*Figura 30. Duplicar un diario del gestor de colas*

El diario local de QM1 en el host ALPHA se llama AMQAJRN (o con su nombre completo, QMQM1/AMQAJRN) y en BETA el diario es QMQM1/AMQBJRN. Cada diario local replica los diarios remotos en todas las otras instancias del gestor de colas. Si el gestor de colas se configura con dos instancias, un diario local se replica en un diario remoto.

### **Réplica de diarios remotos \*SYNC o \*ASYN**

Los diarios de IBM i se duplican mediante el registro por diario síncrono (\*SYNC) o asíncrono (\*ASYN); consulte [Gestión de diarios remotos](#).

El modo de réplica en [Figura 30 en la página 407](#) es \*SYNC, no \*ASYN. \*ASYN es más rápido, pero si se produce una anomalía cuando el estado del diario remoto es \*ASYNPEND, el diario local y remoto no son coherentes. El diario remoto debe ponerse al día con el diario local. Si elige \*SYNC, a continuación, el sistema local espera al diario remoto antes de retroceder desde una llamada que necesite una escritura completa. Los diarios local y remoto permanecen coherentes el uno con el otro. Sólo si la operación \*SYNC tarda más de un tiempo designado<sup>1</sup> y el registro por diario remoto se desactiva, las revistas se desincronizan. Se registra un error en la cola de mensajes de diario y en QSYSOPR. El gestor de colas detecta este mensaje, escribe un error en el registro cronológico de errores del gestor de colas

y desactiva la réplica remota del diario del gestor de colas. La instancia del gestor de colas activo se reanuda sin que se realice un registro por diario remoto en este diario. Cuando el servidor remoto vuelve a estar disponible, debe reactivar manualmente la réplica síncrona del diario remoto. Los diarios se vuelven a sincronizar.

Un problema con la configuración \*SYNC/\*SYNC ilustrado en [Figura 30 en la página 407](#) es cómo toma el control la instancia del gestor de colas en espera en BETA. Tan pronto como la instancia del gestor de colas en BETA escribe su primer mensaje persistente, intenta actualizar el diario remoto en ALPHA. Si la causa de que el control pase de ALPHA a BETA era la anomalía de ALPHA y ALPHA sigue sin funcionar, el registro por diario remoto en ALPHA falla. BETA espera a que ALPHA responda y después desactiva el registro por diario remoto y sigue procesando mensajes sólo con el registro por diario local. BETA debe esperar un poco para detectar que ALPHA no está funcionando, lo que provoca un período de inactividad.

La opción de establecer un registro por diario remoto en \*SYNC o \*ASYNC es cuestión de negociación. [Tabla 29 en la página 408](#) resumen las negociaciones entre un par de gestores de colas sobre la utilización de un registro por diario \*SYNC y \*ASYNC:

Activo	En espera	*SYNC	*ASYNC
*SYNC		<ol style="list-style-type: none"> <li>1. Cambio y migración tras error coherentes</li> <li>2. La instancia en espera no se reanuda inmediatamente después de una migración tras error.</li> <li>3. El registro por diario remoto debe estar disponible todo el tiempo</li> <li>4. El rendimiento del gestor de colas depende del registro por diario remoto</li> </ol>	<ol style="list-style-type: none"> <li>1. Cambio y migración tras error coherentes</li> <li>2. El registro por diario remoto se debe conmutar a *SYNC cuando el servidor en espera está disponible</li> <li>3. El registro por diario remoto debe permanecer disponible después de reiniciarse</li> <li>4. El rendimiento del gestor de colas depende del registro por diario remoto</li> </ol>
*ASYNC		<ol style="list-style-type: none"> <li>1. No es una combinación sensible</li> </ol>	<ol style="list-style-type: none"> <li>1. Algunos mensajes se pueden perder o duplicar después de una sustitución por anomalía o una conmutación</li> <li>2. La instancia en espera no puede estar disponible todo el tiempo para que instancia activa continúe sin retardo.</li> <li>3. El rendimiento no depende del registro por diario remoto</li> </ol>

#### \*SYNC / \*SYNC

Las instancia del gestor de colas activo utiliza el registro por diario \*SYNC y cuando se inicia la instancia del gestor de colas en espera, intenta inmediatamente utilizar el registro por diario \*SYNC.

1. El diario remoto es transaccionalmente coherente con el diario local del gestor de colas activo. Si el gestor de colas se cambia a la instancia en espera, se puede reanudar inmediatamente. La instancia en espera normalmente se reanuda sin ninguna pérdida o duplicación de mensajes. Los mensajes sólo se pierden o se duplican si el registro por diario remoto ha fallado desde el último punto de comprobación y el gestor de colas activo anteriormente no se puede reiniciar.

<sup>1</sup> El tiempo designado es de 60 segundos en IBM i 5 y en el rango de 1 a 3600 segundos en IBM i 6.1 en adelante.



2. Si el gestor de colas pasa a la instancia en espera tras un error, no se podrá iniciar inmediatamente. La instancia del gestor de colas en espera se activa con el registro por diario \*SYNC. La causa de la sustitución por anomalía puede impedir el registro por diario remoto en el servidor que alberga la instancia en espera. El gestor de colas espera hasta que se detecta el problema antes de procesar los mensajes persistentes. Se registra un error en la cola de mensajes de diario y en QSYSOPR. El gestor de colas detecta este mensaje, escribe un error en el registro cronológico de errores del gestor de colas y desactiva la réplica remota del diario del gestor de colas. La instancia del gestor de colas activo se reanuda sin que se realice un registro por diario remoto en este diario. Cuando el servidor remoto vuelve a estar disponible, debe reactivar manualmente la réplica síncrona del diario remoto. Los diarios se vuelven a sincronizar.
3. El servidor en el que se ha replicado el diario remoto siempre debe estar disponible para mantener el diario remoto. El diario remoto se replica normalmente en el mismo servidor que alberga el gestor de colas en espera. Es posible que el servidor no esté disponible. Se registra un error en la cola de mensajes de diario y en QSYSOPR. El gestor de colas detecta este mensaje, escribe un error en el registro cronológico de errores del gestor de colas y desactiva la réplica remota del diario del gestor de colas. La instancia del gestor de colas activo se reanuda sin que se realice un registro por diario remoto en este diario. Cuando el servidor remoto vuelve a estar disponible, debe reactivar manualmente la réplica síncrona del diario remoto. Los diarios se vuelven a sincronizar.
4. El registro por diario remoto es más lento que el registro por diario local y sustancialmente más lento si los servidores están separados por una larga distancia. El gestor de colas debe esperar al registro por diario remoto, lo que reduce el rendimiento del gestor de colas.

La configuración \*SYNC/\*SYNC entre un par de servidores tiene la desventaja de que se produce un retardo al reanudar la instancia en espera después de la sustitución por anomalía. La configuración \*SYNC/\*ASYNCR no tiene este problema.

\*SYNC/\*SYNC garantiza que no se pierdan mensajes después de la conmutación o la sustitución por anomalía mientras haya un diario remoto disponible. Si desea reducir el riesgo de la pérdida de mensajes después del cambio o la migración tras error, tiene dos opciones. Detenga la instancia activa si el diario remoto se vuelve inactivo o cree diarios remotos en más de un servidor.

#### **\*SYNC / \*ASYNCR**

La instancia del gestor de colas activo utiliza el registro por diario \*SYNC y cuando la instancia del gestor de colas en espera se inicia, utiliza el registro por diario \*ASYNCR. Poco después de que el servidor que alberga la instancia en espera nueva esté disponible, el operador del sistema debe conmutar el diario remoto en la instancia activa a \*SYNC. Cuando el operador cambia el registro por diario remoto de \*ASYNCR a \*SYNC, la instancia activa se detiene si el estado del diario remoto es \*ASYNCRPEND. La instancia del gestor de colas activo espera hasta que las entradas de diario restantes se transfieren al diario remoto. Cuando el diario remoto se ha sincronizado con el diario local, la espera nueva vuelve a ser transaccionalmente coherente con la instancia activa nueva. Desde la perspectiva de los gestores de colas multiinstancia, en una configuración \*SYNC/\*ASYNCR, el operador del sistema de IBM i tiene una tarea adicional. El operador debe conmutar el registro de diario remoto a \*SYNC además de reiniciar la instancia del gestor de colas que ha sufrido una anomalía.

1. El diario remoto es transaccionalmente coherente con el diario local del gestor de colas activo. Si la instancia del gestor de colas activa se ha conmutado o falla en la instancia en espera, la instancia en espera se puede reanudar inmediatamente. La instancia en espera normalmente se reanuda sin ninguna pérdida o duplicación de mensajes. Los mensajes sólo se pierden o se duplican si el registro por diario remoto ha fallado desde el último punto de comprobación y el gestor de colas activo anteriormente no se puede reiniciar.
2. El operador del sistema debe cambiar el diario remoto de \*ASYNCR a \*SYNC poco después de que el sistema que alberga la instancia activa vuelva a estar disponible. El operador puede esperar a que el diario remoto se ponga al día antes de conmutar el diario remoto a \*SYNC. El operador también puede conmutar inmediatamente la instancia remota a \*SYNC y forzar que la instancia activa espere a que el diario de instancia en espera se haya puesto al día. Cuando el registro por diario remoto se establece en \*SYNC, la instancia en espera es generalmente transaccionalmente coherente con la instancia activa. Los mensajes sólo se pierden o se duplican si el registro por diario remoto ha fallado desde el último punto de comprobación y el gestor de colas activo anteriormente no se puede reiniciar.

3. Cuando la configuración se ha restaurado de una conmutación o una sustitución por anomalía, el servidor en el que el diario remoto está alojado debe estar disponible todo el tiempo.

Elija \*SYNC/\*ASYNCR cuando desee que el gestor de colas en espera se reanude rápidamente después de una sustitución por anomalía. Debe restaurar manualmente el valor del diario remoto a \*SYNC en la instancia activa nueva. La configuración \*SYNC/\*ASYNCR coincide con el patrón normal de administrar un par de gestores de colas multiinstancia. Cuando una instancia sufre una anomalía, hay un tiempo antes de reiniciar la instancia en espera durante el que no se puede realizar una migración tras error de la instancia activa.

#### **\*ASYNCR / \*ASYNCR**

Los servidores que albergan los gestores de colas activo y en espera están configurados para utilizar el registro por diario remoto \*ASYNCR.

1. Cuando se producen la conmutación o la sustitución por anomalía, el gestor de colas sigue con el diario en el servidor nuevo. El diario no se puede sincronizar cuando tiene lugar la conmutación o sustitución por anomalía. En consecuencia, los mensajes se pueden perder o duplicar.
2. La instancia activa se ejecuta incluso aunque el servidor que alberga el gestor de colas en espera no esté disponible. El diario local se replica asíncronamente con el servidor en espera cuando está disponible.
3. El rendimiento del gestor de colas local no se ve afectado por el registro por diario remoto.

Elija \*ASYNCR/\*ASYNCR si el rendimiento es su principal requisito y si está preparado para perder o duplicar algunos mensajes después de la conmutación o la sustitución por anomalía.

#### **\*ASYNCR / \*SYNC**

No hay razón para utilizar esta combinación de opciones.

### **Activación del gestor de colas desde un diario remoto**

Los diarios se pueden replicar síncrona o asíncronamente. Es posible que el diario remoto no esté activo, o es posible que se esté actualizando con el diario local. Puede que el diario remoto se esté actualizando incluso aunque se esté replicando de forma síncrona, porque se haya activado recientemente. Las reglas que el gestor de colas aplica al estado del diario remoto que utiliza durante el inicio, son las siguientes.

1. El inicio en espera falla si debe repetirse desde el diario remoto en el recurso en espera y el estado del diario es \*FAILED o \*INACTPEND.
2. Cuando empieza la activación del recurso en espera, el estado del diario remoto en espera debe ser \*ACTIVE o \*INACTIVE. Si el estado es \*INACTIVE, es posible que la activación falle si no se han replicado todos los datos del diario.

La anomalía se produce si los datos del gestor de colas del sistema de archivos de red tiene un registro de punto de comprobación más reciente que el presente en el diario remoto. La anomalía no se puede producirá mientras el diario remoto se active dentro del intervalo máximo de 30 minutos predeterminado entre puntos de comprobación. Si el gestor de colas en espera lee un registro de punto de comprobación más reciente del sistema de archivos, no se inicia.

Tiene una opción: espere a que se pueda restaurar el diario local en el servidor activo o realice un inicio en frío del gestor de colas en espera. Si opta por el inicio en frío, el gestor de colas se inicia sin datos de diario y confía en la coherencia y la integridad de los datos del gestor de colas del sistema de archivos.

**Nota:** Si realiza un inicio frío de un gestor de colas, corre el riesgo de perder o duplicar mensajes después del último punto de comprobación. Las transacciones de mensaje se escribieron en el diario pero es posibles que algunas de ellas no se hayan escrito en los datos del gestor de colas en el sistema de archivos. Cuando realiza un inicio frío de un gestor de colas, se inicia un diario nuevo y las transacciones que no se escriben en los datos del gestor de colas del sistema de archivos se pierden.

3. La activación del gestor de colas en espera a que el estado del diario remoto en espera cambie de \*ASYNCRPEND o \*SYNCRPEND a \*ASYNCR o \*SYNC. Los mensajes se escriben en el registro de trabajo del controlador de ejecución de forma periódica.

**Nota:** En este caso, la activación espera en el diario remoto local al gestor de colas en espera que se está activando. El gestor de colas espera también un tiempo antes de continuar sin un diario remoto. Espera cuando intenta escribir de forma síncrona en su diario remoto (o diarios) y el diario no está disponible.

4. La activación se detendrá cuando el estado del diario cambie a \*FAILED o \*INACTPEND.

Los nombres y los estados de los diarios locales y remotos que deben utilizarse en la activación se escriben en el registro cronológico de errores del gestor de colas.

### Creación de un gestor de colas multiinstancia mediante la duplicación de diarios y NetServer en IBM i

Cree un gestor de colas multiinstancia para ejecutarlo en dos servidores IBM i. Los datos del gestor de colas almacenan en un tercer servidor IBM i en el que se utiliza NetServer. El diario del gestor de colas se duplica entre los dos servidores mediante el registro por diario remoto. El mandato **ADDQMJRN** se utiliza para simplificar la creación de los diarios remotos.

## Antes de empezar

1. La tarea requiere tres servidores IBM i. Instale IBM MQ en dos de ellos, ALPHA y BETA, en el ejemplo. IBM MQ debe estar, como mínimo, en la versión 7.0.1.1.
2. El tercer servidor es un servidor IBM i, conectado mediante NetServer a ALPHA y BETA. Se utiliza para compartir los datos del gestor de colas. No es necesario que tenga una instalación de IBM MQ. Es útil instalar IBM MQ en el servidor como un paso temporal para configurar los directorios del gestor de colas y los permisos.
3. Asegúrese de que el perfil de usuario de QMQM tenga la misma contraseña en los tres servidores.
4. Instale IBM i NetServer; consulte [i5/OS NetServer](#).

## Acerca de esta tarea

Siga estos pasos para crear la configuración descrita en la [Figura 31 en la página 414](#). Los datos del gestor de colas se conectan mediante IBM i NetServer.

- Cree conexiones desde ALPHA y BETA a la unidad compartida de directorios de GAMMA que va a almacenar los datos del gestor de colas. La tarea también configura los permisos necesarios, los perfiles de usuario y las contraseñas.
- Añada entradas de base de datos relacional (RDBE) a los sistemas IBM i en los que se vayan a ejecutar las instancias del gestor de colas. Las entradas RDBE se utilizan para conectar con los sistemas IBM i que se utilizan para el registro por diario remoto.
- Cree el gestor de colas QM1 en el servidor IBM i, ALPHA.
- Añada la información de control de gestor de colas para QM1 en el otro servidor IBM i, BETA.
- Cree diarios remotos en los servidores IBM i para ambas instancias del gestor de colas. Cada gestor de colas escribe en el diario local. El diario local se replica en el diario remoto. El mandato **ADDQMJRN** simplifica la adición de diarios y las conexiones.
- Inicie el gestor de colas, permitiendo una instancia en espera.

## Procedimiento

1. Lleve a cabo la tarea “[Creación de una unidad compartida de datos de un gestor de colas utilizando NetServer en IBM i](#)” en la página 398.

Como resultado, ALPHA y BETA tienen una participación, /QNTC/GAMMA/WMQ, que apunta a /QIBM/UserData/mqm/qmgrs en GAMMA. Los perfiles de usuario QMQM y QMQMADM tienen los permisos necesarios y QMQM tiene contraseñas coincidentes en los tres sistemas.

2. Añada entradas de base de datos relacional (RDBE) a los sistemas IBM i en los que se vayan a alojar las instancias del gestor de colas.

a) En ALPHA cree la conexión con BETA.

```
ADDRDBDIRE RDB(BETA) RMTLOCNAME(BETA *IP) RMTAUTMTH(*USRIDPWD)
```

b) En BETA cree las conexiones con ALPHA.

```
ADDRDBDIRE RDB(ALPHA) RMTLOCNAME(ALPHA *IP) RMTAUTMTH(*USRIDPWD)
```

3. Cree el gestor de colas QM1 en ALPHA, guardando los datos del gestor de colas en GAMMA.

```
CRTMQM MQMNAME(QM1) UDLMSGQ(SYSTEM.DEAD.LETTER.QUEUE)
MQMDIRP(' /QNTC/GAMMA/WMQ')
```

La vía de acceso, utiliza NetServer para crear los datos del gestor de colas.

4. Ejecutar en ALPHA. El mandato añade un diario remoto en BETA para.

```
ADDMQJRN MQMNAME(QM1) RMTJRNRDB(BETA)
```

crea entradas de diario en su diario local en ALPHA cuando la instancia activa de está en ALPHA. El diario local en ALPHA se replica con el diario remoto en BETA.

5. Utilice el mandato,, para inspeccionar los datos de configuración de IBM MQ creados por en ALPHA.

La información se necesita en el paso siguiente.

En este ejemplo, se crea la siguiente configuración en ALPHA para:

```
Name=QM1
Prefix=/QIBM/UserData/mqm
Library=QMOM1
Directory=QM1
DataPath= /QNTC/GAMMA/WMQ /QM1
```

6. Cree una instancia de gestor de colas de QM1 en BETA utilizando el mandato. Ejecute el mandato siguiente en BETA para modificar la información de control del gestor de colas en BETA.

```
ADDMQMINF MQMNAME(QM1)
PREFIX('/QIBM/UserData/mqm')
MQMDIR(QM1)
MQMLIB(QMOM1)
DATAPATH('/QNTC/GAMMA/WMQ /QM1')
```

**Consejo:** Copie y pegue la información de configuración. La stanza del gestor de colas es la misma en ALPHA y BETA.

7. Ejecutar en BETA. El mandato añade un diario local en BETA y un diario remoto en ALPHA para.

```
ADDMQJRN MQMNAME(QM1) RMTJRNRDB(ALPHA)
```

crea entradas de diario en su diario local en BETA cuando la instancia activa de está en BETA. El diario local en BETA se replica con el diario remoto en ALPHA.

**Nota:** Como alternativa, puede configurar el registro por diario remoto de BETA a ALPHA mediante el registro por diario asíncrono.

Utilice este mandato para configurar el registro por diario asíncrono de BETA a ALPHA, en lugar del mandato en el paso “7” en la [página 412](#).

```
ADDMQJRN MQMNAME(QM1) RMTJRNRDB(ALPHA) RMTJRNDLV(*ASYNCR)
```

Si el servidor o el registro por diario en ALPHA es el origen de la anomalía, BETA se inicia sin esperar a que se repliquen entradas del diario nuevas en ALPHA.

Cambie la modalidad de réplica a \*SYNC, utilizando el mandato, cuando ALPHA vuelva a estar en línea.

Utilice la información en [“Configuración de un diario duplicado para ASP en IBM i”](#) en la página 406 para decidir si desea duplicar los diarios síncrona o asíncronamente o utilizar una mezcla de ambas modalidades. El valor predeterminado consiste en replicar síncronamente con un período de espera de 60 segundos para obtener una respuesta del diario remoto.

8. Verifique que los diarios en ALPHA y BETA estén habilitados y que el estado de la réplica de diario remoto sea.

a) En ALPHA:

```
WRKMQMJRN MQMNAME(QM1)
```

b) En BETA:

```
WRKMQMJRN MQMNAME(QM1)
```

9. Inicie las instancias del gestor de colas en ALPHA y BETA.

a) Inicie la primera instancia en ALPHA, convirtiéndola en la instancia activa. Habilite la conmutación a una instancia en espera.

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

b) Inicie la segunda instancia en BETA, convirtiéndola en la instancia en espera.

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

## Resultados

Se utiliza para comprobar el estado del gestor de colas:

1. El estado de la instancia del gestor de colas en ALPHA debe ser.
2. El estado de la instancia del gestor de colas en BETA debe ser.

## Ejemplo

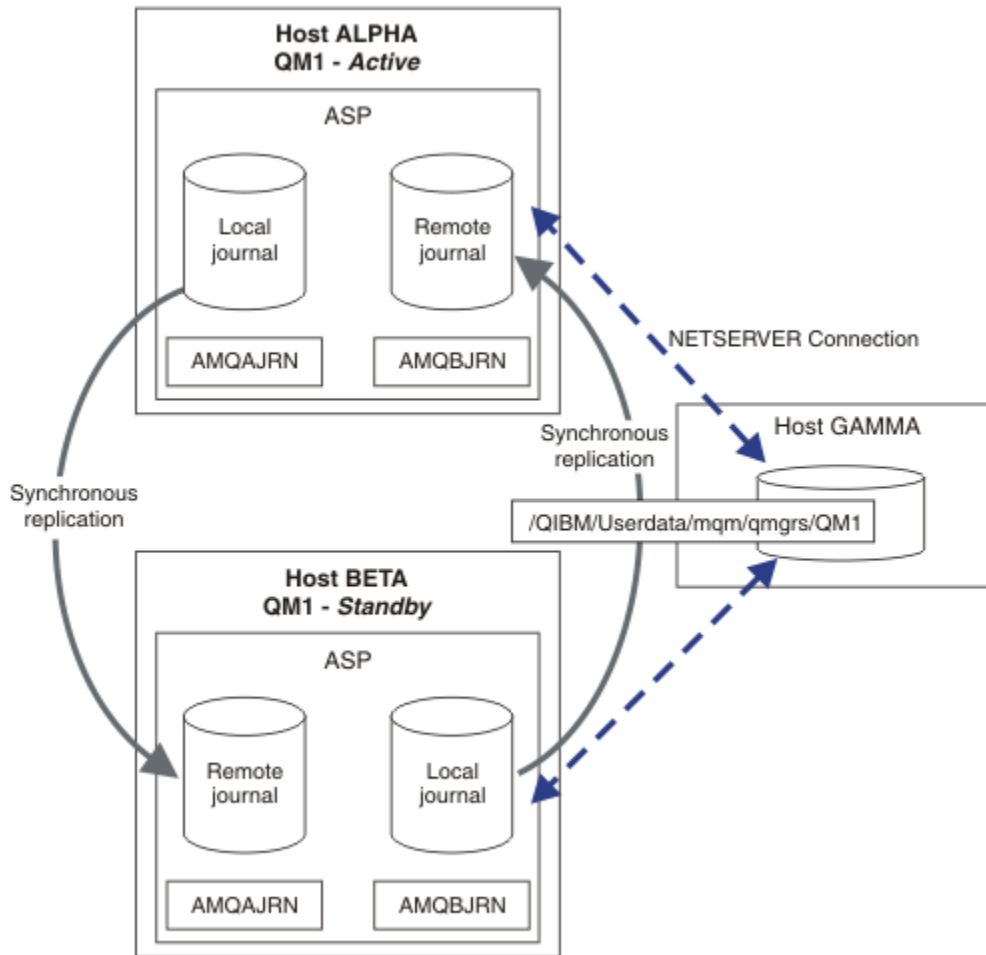


Figura 31. Configuración de diario duplicada

## Qué hacer a continuación

- Verifique que las instancias activa y en espera se conmutan automáticamente. Puede ejecutar los programas de alta disponibilidad de muestra para probar la conmutación; consulte el tema [Programas de ejemplo de alta disponibilidad](#). Los programas de ejemplo son clientes 'C'. Puede ejecutarlos desde una plataforma Windows o Unix.

1. Inicie los programas de ejemplo de alta disponibilidad.
2. En ALPHA, finalice el gestor de colas solicitando una conmutación:

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) ALSWITCH(*YES)
```

3. Compruebe que la instancia de en BETA esté activa.
4. Reiniciar en ALPHA

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

- Busque configuraciones de alta disponibilidad alternativas:

1. Utilice NetServer para colocar los datos del gestor de colas en un servidor Windows.

2. En lugar de utilizar el registro por diario remoto para duplicar el diario del gestor de colas, almacene el diario en una ASP independiente. Utilice la agrupación en clúster de IBM i para transferir la ASP independiente de ALPHA a BETA.

### **IBM i** *Conversión de un gestor de colas de instancia única en un gestor de colas multiinstancia usando NetServer y duplicación de diarios en IBM i*

Convierta un gestor de colas de una sola instancia en un gestor de colas multiinstancia. Mueva los datos del gestor de colas a una unidad compartida de red conectada por NetServer. Duplique el diario del gestor de colas en un segundo servidor IBM i, utilizando el registro por diario remoto.

#### **Antes de empezar**

1. La tarea requiere tres servidores IBM i. La instalación existente de IBM MQ, en el servidor ALPHA en el ejemplo, debe ser como mínimo de IBM WebSphere MQ 7.0.1 Fix Pack 1. ALPHA está ejecutando un gestor de colas llamado QM1 en el ejemplo.
2. Instale IBM MQ en el segundo servidor IBM i, BETA, en el ejemplo.
3. El tercer servidor es un servidor IBM i, conectado mediante NetServer a ALPHA y BETA. Se utiliza para compartir los datos del gestor de colas. No es necesario que tenga una instalación de IBM MQ. Es útil instalar IBM MQ en el servidor como un paso temporal para configurar los directorios del gestor de colas y los permisos.
4. Asegúrese de que el perfil de usuario de QMQM tenga la misma contraseña en los tres servidores.
5. Instale IBM i NetServer; consulte [i5/OS NetServer](#).

#### **Acerca de esta tarea**

Realice los pasos siguientes para convertir un gestor de colas de una sola instancia en el gestor de colas multiinstancia que se muestra en la [Figura 32 en la página 419](#). El gestor de colas de instancia única se borra en la tarea y después se vuelve a crear, almacenando los datos del gestor de colas en la unidad compartida de red conectada por NetServer. Este procedimiento es más fiable que mover los directorios del gestor de colas y los archivos a la unidad compartida de red mediante el mandato **CPY**.

- Cree conexiones desde ALPHA y BETA a la unidad compartida de directorios de GAMMA que va a almacenar los datos del gestor de colas. La tarea también configura los permisos necesarios, los perfiles de usuario y las contraseñas.
- Añada entradas de base de datos relacional (RDBE) a los sistemas IBM i en los que se vayan a ejecutar las instancias del gestor de colas. Las entradas RDBE se utilizan para conectar con los sistemas IBM i que se utilizan para el registro por diario remoto.
- Guarde los registros y las definiciones del gestor de colas, detenga el gestor de colas y suprimalo.
- Vuelva a crear el gestor de colas, almacenando los datos del gestor de colas en la unidad compartida de red en GAMMA.
- Añada la segunda instancia del gestor de colas al otro servidor.
- Cree diarios remotos en los servidores IBM i para ambas instancias del gestor de colas. Cada gestor de colas escribe en el diario local. El diario local se replica en el diario remoto. El mandato **ADDQMJRN** simplifica la adición de diarios y las conexiones.
- Inicie el gestor de colas, permitiendo una instancia en espera.

#### **Nota:**

En el paso “4” en la [página 416](#) de la tarea, suprime el gestor de colas de una sola instancia, QM1. Al suprimir el gestor de colas se suprimen todos los mensajes persistentes en las colas. Por esta razón, complete el proceso de todos los mensajes almacenados por el gestor de colas antes de convertirlo. Si no es posible procesar todos los mensajes, haga una copia de seguridad de la biblioteca del gestor de colas antes de “4” en la [página 416](#). Restablezca la biblioteca del gestor de colas después del paso “5” en la [página 416](#).

#### **Nota:**

En el paso “5” en la página 416 de la tarea, vuelva a crear QM1. Aunque el gestor de colas tenga el mismo nombre, tiene un identificador de gestor de colas diferente. La agrupación en clúster del gestor de colas utiliza el identificador del gestor de colas. Para suprimir y volver a crear un gestor de colas en un clúster, debe eliminar primero el gestor de colas del clúster; consulte [Eliminación de un gestor de colas de un clúster: Método alternativo](#) o [Eliminación de un gestor de colas de un clúster](#). Cuando haya vuelto a crear el gestor de colas, añádalo al clúster. Aunque tiene el mismo nombre que antes, los otros gestores de colas del clúster lo consideran un gestor de colas nuevo.

## Procedimiento

1. Lleve a cabo la tarea [“Creación de una unidad compartida de datos de un gestor de colas utilizando NetServer en IBM i”](#) en la página 398.

Como resultado, ALPHA y BETA tienen una participación, /QNTC/GAMMA/WMQ, que apunta a /QIBM/UserData/mqm/qmgrs en GAMMA. Los perfiles de usuario QMQM y QMQMADM tienen los permisos necesarios y QMQM tiene contraseñas coincidentes en los tres sistemas.

2. Añada entradas de base de datos relacional (RDBE) a los sistemas IBM i en los que se vayan a alojar las instancias del gestor de colas.

- a) En ALPHA cree la conexión con BETA.

```
ADDRDBDIRE RDB(BETA) RMTLOCNAME(BETA *IP) RMTAUTMTH(*USRIDPWD)
```

- b) En BETA cree las conexiones con ALPHA.

```
ADDRDBDIRE RDB(ALPHA) RMTLOCNAME(ALPHA *IP) RMTAUTMTH(*USRIDPWD)
```

3. Cree los scripts que vuelven a crear los objetos del gestor de colas.

```
QSAVEQMGR LCLQMGRNAM(QM1) FILENAME('*CURLIB/QMQSC(QM1)')
OUTPUT(*REPLACE) MAKEAUTH(*YES) AUTHFN('*CURLIB/QMAUT(QM1)')
```

4. Detenga el gestor de colas y suprimalo.

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) ENDCCTJOB(*YES) RCDMQMIMG(*YES) TIMEOUT(15)
DLTMQM MQMNAME(QM1)
```

5. Cree el gestor de colas QM1 en ALPHA, guardando los datos del gestor de colas en GAMMA.

```
CRTMQM MQMNAME(QM1) UDLMSGQ(SYSTEM.DEAD.LETTER.QUEUE)
MQMDIRP('/QNTC/GAMMA/WMQ')
```

La vía de acceso, utiliza NetServer para crear los datos del gestor de colas.

6. Vuelva a crear los objetos del gestor de colas QM1 a partir de las definiciones guardadas.

```
STRMQMQSC SRCMBR(QM1) SRCFILE(*CURLIB/QMQSC) MQMNAME(QM1)
```

7. Aplique las autorizaciones de la información guardada.

- a) Compile el programa de autorización guardado.

```
CRTCLPGM PGM(*CURLIB/QM1) SRCFILE(*CURLIB/QMAUT)
SRCMBR(QM1) REPLACE(*YES)
```

- b) Ejecute el programa para aplicar las autorizaciones.

```
CALL PGM(*CURLIB/QM1)
```



c) Renueve la información de seguridad para QM1.

```
RFRMQMAUT MQMNAME(QM1)
```

8. Ejecutar en ALPHA. El mandato añade un diario remoto en BETA para.

```
ADDMQMJRN MQMNAME(QM1) RMTJNRDB(BETA)
```

crea entradas de diario en su diario local en ALPHA cuando la instancia activa de está en ALPHA. El diario local en ALPHA se replica con el diario remoto en BETA.

9. Utilice el mandato,, para inspeccionar los datos de configuración de IBM MQ creados por en ALPHA.

La información se necesita en el paso siguiente.

En este ejemplo, se crea la siguiente configuración en ALPHA para:

```
Name=QM1
Prefix=/QIBM/UserData/mqm
Library=QMOM1
Directory=QM1
DataPath= /QNTC/GAMMA/WMQ /QM1
```

10. Cree una instancia de gestor de colas de QM1 en BETA utilizando el mandato. Ejecute el mandato siguiente en BETA para modificar la información de control del gestor de colas en BETA.

```
ADDMQMINF MQMNAME(QM1)
PREFIX('/QIBM/UserData/mqm')
MQMDIR(QM1)
MQMLIB(QMOM1)
DATAPATH('/QNTC/GAMMA/WMQ /QM1')
```

**Consejo:** Copie y pegue la información de configuración. La stanza del gestor de colas es la misma en ALPHA y BETA.

11. Ejecutar en BETA. El mandato añade un diario local en BETA y un diario remoto en ALPHA para.

```
ADDMQMJRN MQMNAME(QM1) RMTJNRDB(ALPHA)
```

crea entradas de diario en su diario local en BETA cuando la instancia activa de está en BETA. El diario local en BETA se replica con el diario remoto en ALPHA.

**Nota:** Como alternativa, puede configurar el registro por diario remoto de BETA a ALPHA mediante el registro por diario asíncrono.

Utilice este mandato para configurar el registro por diario asíncrono de BETA a ALPHA, en lugar del mandato en el paso “7” en la [página 412](#).

```
ADDMQMJRN MQMNAME(QM1) RMTJNRDB(ALPHA) RMTJRNDLV(*ASYNCR)
```

Si el servidor o el registro por diario en ALPHA es el origen de la anomalía, BETA se inicia sin esperar a que se repliquen entradas del diario nuevas en ALPHA.

Cambie la modalidad de réplica a \*SYNCR, utilizando el mandato, cuando ALPHA vuelva a estar en línea.

Utilice la información en “Configuración de un diario duplicado para ASP en IBM i” en la [página 406](#) para decidir si desea duplicar los diarios síncrona o asíncronamente o utilizar una mezcla de ambas modalidades. El valor predeterminado consiste en replicar síncronamente con un período de espera de 60 segundos para obtener una respuesta del diario remoto.

12. Verifique que los diarios en ALPHA y BETA estén habilitados y que el estado de la réplica de diario remoto sea.

a) En ALPHA:

```
WRKMQMJRN MQMNAME(QM1)
```

b) En BETA:

```
WRKMQMJRN MQMNAME(QM1)
```

13. Inicie las instancias del gestor de colas en ALPHA y BETA.

a) Inicie la primera instancia en ALPHA, convirtiéndola en la instancia activa. Habilite la conmutación a una instancia en espera.

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

b) Inicie la segunda instancia en BETA, convirtiéndola en la instancia en espera.

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

## Resultados

Se utiliza para comprobar el estado del gestor de colas:

1. El estado de la instancia del gestor de colas en ALPHA debe ser.
2. El estado de la instancia del gestor de colas en BETA debe ser.

## Ejemplo

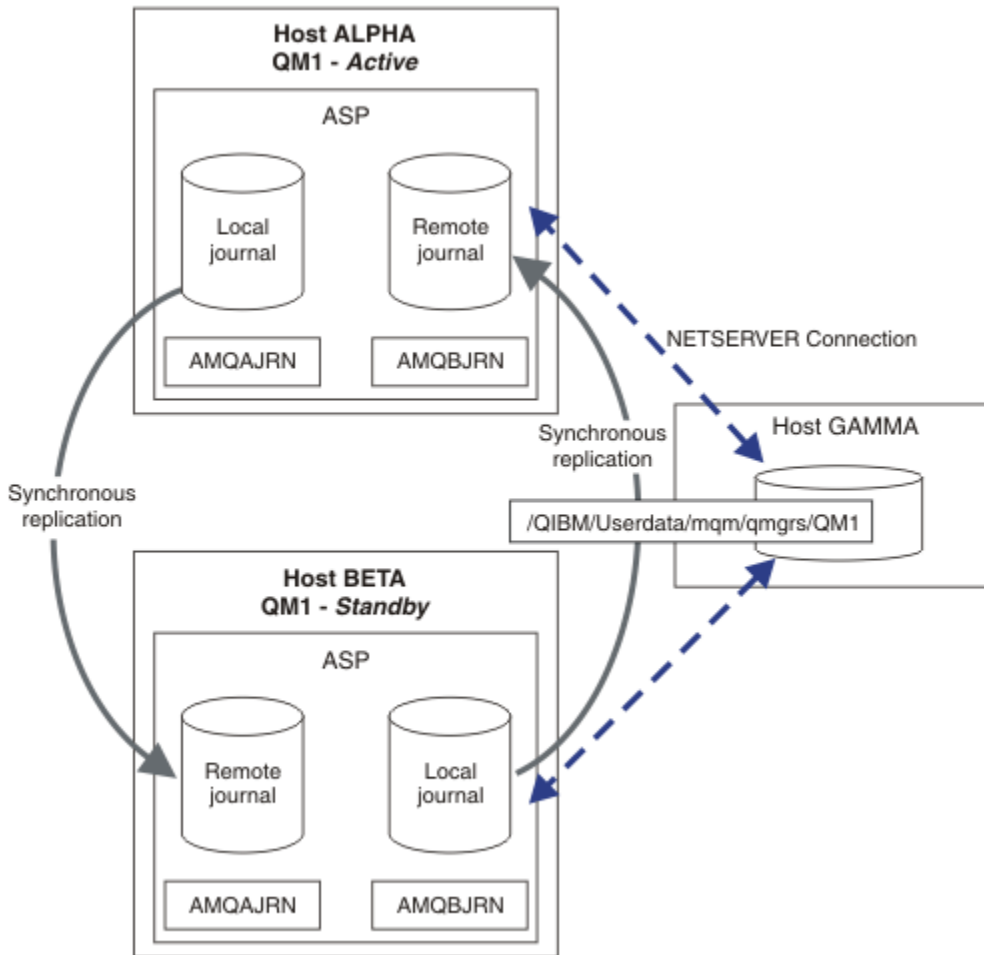


Figura 32. Configuración de diario duplicada

## Qué hacer a continuación

- Verifique que las instancias activa y en espera se conmutan automáticamente. Puede ejecutar los programas de alta disponibilidad de muestra para probar la conmutación; consulte el tema [Programas de ejemplo de alta disponibilidad](#). Los programas de ejemplo son clientes 'C'. Puede ejecutarlos desde una plataforma Windows o Unix.

1. Inicie los programas de ejemplo de alta disponibilidad.
2. En ALPHA, finalice el gestor de colas solicitando una conmutación:

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) ALSWITCH(*YES)
```

3. Compruebe que la instancia de en BETA esté activa.
4. Reiniciar en ALPHA

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

- Busque configuraciones de alta disponibilidad alternativas:

1. Utilice NetServer para colocar los datos del gestor de colas en un servidor Windows.

2. En lugar de utilizar el registro por diario remoto para duplicar el diario del gestor de colas, almacene el diario en una ASP independiente. Utilice la agrupación en clúster de IBM i para transferir la ASP independiente de ALPHA a BETA.

### IBM i Configuración de diario ASP independiente conmutada en IBM i

No es necesario que duplique un diario ASP independiente para crear una configuración de gestor de colas multiinstancia. Debe automatizar un medio para transferir la ASP independiente desde el gestor de colas activo al gestor de colas en espera. Hay soluciones alternativas de alta disponibilidad que se pueden aplicar con una ASP independiente y no todas ellas requieren el uso de un gestor de colas multiinstancia.

Cuando utilice una ASP independiente, no es necesario que duplique el diario de gestor de colas. Si ha instalado la gestión de clústeres y los servidores que hospedan las instancias del gestor de colas se encuentran en el mismo grupo de recursos de clúster, a continuación, el diario del gestor de colas se podrá transferir automáticamente a otro servidor que quede dentro de una distancia corta del servidor activo, si el host que ejecuta la instancia activa falla. También puede transferir el diario manualmente, como parte de una conmutación planificada, o puede escribir un procedimiento de mandato para transferir la ASP independiente programáticamente.

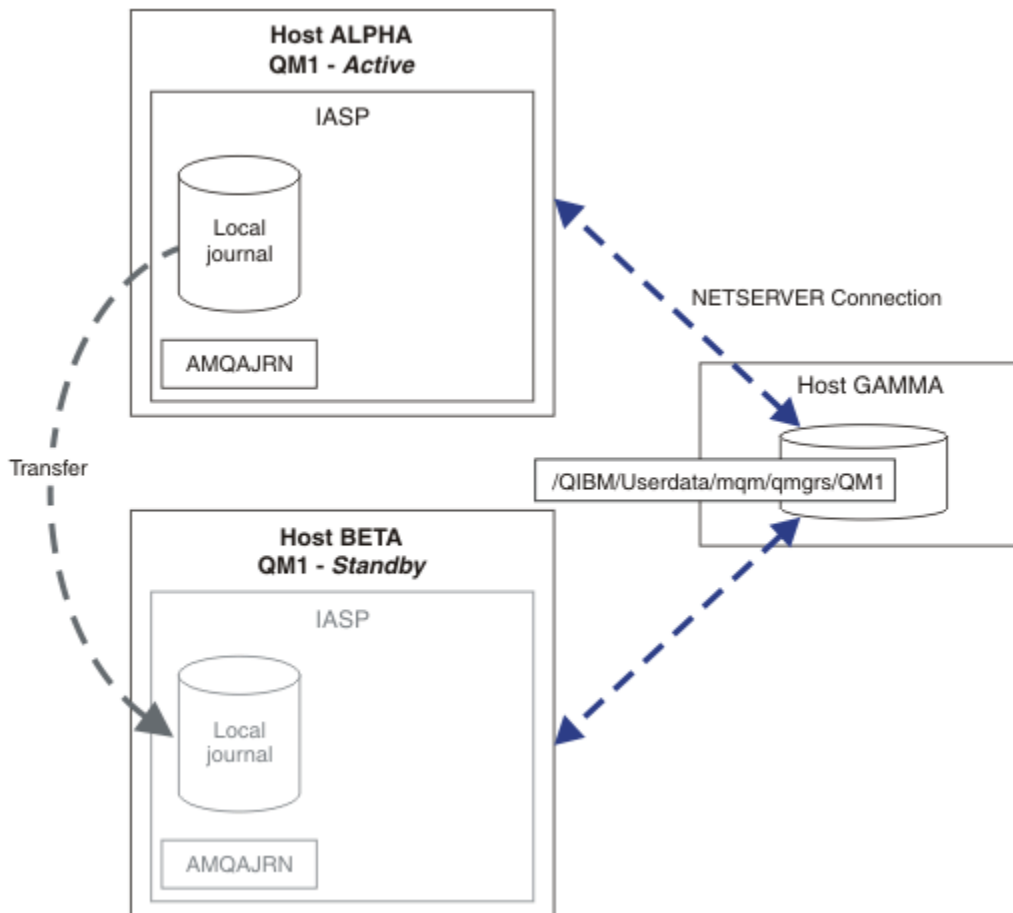


Figura 33. Transferir un diario de gestor de colas utilizando la ASP independiente

Para la operación del gestor de colas multiinstancia, los datos del gestor de colas deben almacenarse en un sistema de archivos compartido. El sistema de archivos puede estar alojado en una gran variedad de plataformas distintas. No puede almacenar datos del gestor de colas multiinstancia en una ASP ni en una ASP independiente.

El sistema de archivos compartidos desempeña dos roles en la configuración: los mismos datos del gestor de colas se comparten entre todas las instancias del gestor de colas. El sistema de archivos tiene

un protocolo de bloqueo robusto que garantiza que sólo una instancia del gestor de colas pueda acceder a los datos del gestor de colas una vez éste se haya iniciado. Si el gestor de colas falla o se interrumpen las comunicaciones con el servidor de archivos, el sistema de archivos debe liberar el bloqueo de los datos del gestor de colas mantenido por la instancia activa que ya no se puede comunicarse con el servidor de archivos. La instancia del gestor de colas en espera puede entonces obtener acceso de lectura/escritura a los datos del gestor de colas. El protocolo del sistema de archivos debe ajustarse a un conjunto de reglas para funcionar correctamente con gestores de colas multiinstancia; consulte [“Componentes de una solución de alta disponibilidad en IBM i”](#) en la página 397.

El mecanismo de bloqueo serializa el inicio del mandato del gestor de colas de inicio y controla qué instancia del gestor de colas está activa. Una vez el gestor de colas pasa a estar activo, vuelva a crear sus colas desde el diario local que el usuario o el clúster HA (de alta disponibilidad) ha transferido al servidor en espera. Los clientes que se pueden volver a conectar y que están esperando la reconexión con el mismo gestor de colas se reconectan y todas las transacciones en curso se restituyen. Las aplicaciones que están configuradas para iniciarse como servicios de gestor de colas se inician.

Debe asegurarse de que el diario local de la instancia del gestor de colas activa que haya fallado en la ASP independiente se transfiera al servidor que hospeda la nueva instancia del gestor de colas en espera activada, o bien configurando el gestor de recursos de clúster, o bien transfiriendo la ASP independiente manualmente. La utilización de ASP independientes no excluye la configuración de diarios remotos y de la duplicación, si decide utilizar ASP independientes para la copia de seguridad y la recuperación tras desastre, ni utilizar duplicación de diarios remotos para la configuración de gestores de cola multiinstancia.

Si ha decidido utilizar la ASP independiente, puede tener en cuenta que existen configuraciones alternativas de alta disponibilidad. La información relacionada de estas soluciones se describe en [“ASP independientes y alta disponibilidad”](#) en la página 424.

1. En lugar de utilizar gestores de colas multiinstancia, instale y configure un gestor de colas de una sola instancia exclusivamente en una ASP independiente, y utilice los servicios de alta disponibilidad de IBM i para conmutar el gestor de colas. Probablemente necesitará aumentar la solución con un monitor de gestor de colas para detectar si el gestor de colas ha fallado independientemente del servidor. Esta es la base de la solución que se proporciona en el *Supportpac MC41: Configuring IBM MQ for iSeries for High Availability*.
2. Utilice ASP independientes y la duplicación entre sistemas (XSM) para duplicar la ASP independiente, en lugar de conmutar la ASP independiente en el bus local. Ello permite ampliar la cobertura geográfica de la solución de ASP independiente en lo que se refiere al tiempo necesario para escribir registros de anotaciones a través de una larga distancia.

## Creación de un gestor de colas multiinstancia usando una ASP independiente y NetServer en IBM i

Cree un gestor de colas multiinstancia para ejecutarlo en dos servidores IBM i. Los datos del gestor de colas se almacenan en un servidor IBM i utilizando NetServer. El diario del gestor de colas se almacena en una ASP independiente. Utilice la agrupación en clúster de IBM i o un procedimiento manual para transferir la ASP independiente que contiene el diario del gestor de colas al otro servidor IBM i.

### Antes de empezar

1. La tarea requiere tres servidores IBM i. Instale IBM MQ en dos de ellos, ALPHA y BETA, en el ejemplo. IBM MQ debe estar, como mínimo, en la versión 7.0.1.1.
2. El tercer servidor es un servidor IBM i, conectado mediante NetServer a ALPHA y BETA. Se utiliza para compartir los datos del gestor de colas. No es necesario que tenga una instalación de IBM MQ. Es útil instalar IBM MQ en el servidor como un paso temporal para configurar los directorios del gestor de colas y los permisos.
3. Asegúrese de que el perfil de usuario de QMQM tenga la misma contraseña en los tres servidores.
4. Instale IBM i NetServer; consulte [i5/OS NetServer](#).
5. Cree procedimientos para transferir la ASP independiente desde el gestor de colas fallido al recurso en espera que lo sustituye. En el *SupportPac MC41: Configuring IBM MQ for iSeries for High Availability*

puede encontrar algunas técnicas útiles para diseñar sus procedimientos de transferencia de ASP independiente.

## Acerca de esta tarea

Siga estos pasos para crear la configuración descrita en la [Figura 34 en la página 423](#). Los datos del gestor de colas se conectan mediante IBM i NetServer.

- Cree conexiones desde ALPHA y BETA a la unidad compartida de directorios de GAMMA que va a almacenar los datos del gestor de colas. La tarea también configura los permisos necesarios, los perfiles de usuario y las contraseñas.
- Cree el gestor de colas QM1 en el servidor IBM i, ALPHA.
- Añada la información de control de gestor de colas para QM1 en el otro servidor IBM i, BETA.
- Inicie el gestor de colas, permitiendo una instancia en espera.

## Procedimiento

1. Lleve a cabo la tarea [“Creación de una unidad compartida de datos de un gestor de colas utilizando NetServer en IBM i”](#) en la página 398.

Como resultado, ALPHA y BETA tienen una participación, /QNTC/GAMMA/WMQ, que apunta a /QIBM/UserData/mqm/qmgrs en GAMMA. Los perfiles de usuario QMQM y QMQMADM tienen los permisos necesarios y QMQM tiene contraseñas coincidentes en los tres sistemas.

2. Cree el gestor de colas QM1 en ALPHA, guardando los datos del gestor de colas en GAMMA.

```
CRTMQM MQMNAME(QM1) UDLMSGQ(SYSTEM.DEAD.LETTER.QUEUE)
MQMDIRP(' /QNTC/GAMMA/WMQ ')
```

La vía de acceso, utiliza NetServer para crear los datos del gestor de colas.

3. Utilice el mandato,, para inspeccionar los datos de configuración de IBM MQ creados por en ALPHA.

La información se necesita en el paso siguiente.

En este ejemplo, se crea la siguiente configuración en ALPHA para:

```
Name=QM1
Prefix=/QIBM/UserData/mqm
Library=QMOM1
Directory=QM1
DataPath= /QNTC/GAMMA/WMQ /QM1
```

4. Cree una instancia de gestor de colas de QM1 en BETA utilizando el mandato. Ejecute el mandato siguiente en BETA para modificar la información de control del gestor de colas en BETA.

```
ADDQMINF MQMNAME(QM1)
PREFIX('/QIBM/UserData/mqm')
MQMDIR(QM1)
MQMLIB(QMQM1)
DATAPATH(' /QNTC/GAMMA/WMQ /QM1 ')
```

**Consejo:** Copie y pegue la información de configuración. La stanza del gestor de colas es la misma en ALPHA y BETA.

5. Inicie las instancias del gestor de colas en ALPHA y BETA.

- a) Inicie la primera instancia en ALPHA, convirtiéndola en la instancia activa. Habilite la conmutación a una instancia en espera.

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

- b) Inicie la segunda instancia en BETA, convirtiéndola en la instancia en espera.

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

## Resultados

Se utiliza para comprobar el estado del gestor de colas:

1. El estado de la instancia del gestor de colas en ALPHA debe ser.
2. El estado de la instancia del gestor de colas en BETA debe ser.

## Ejemplo

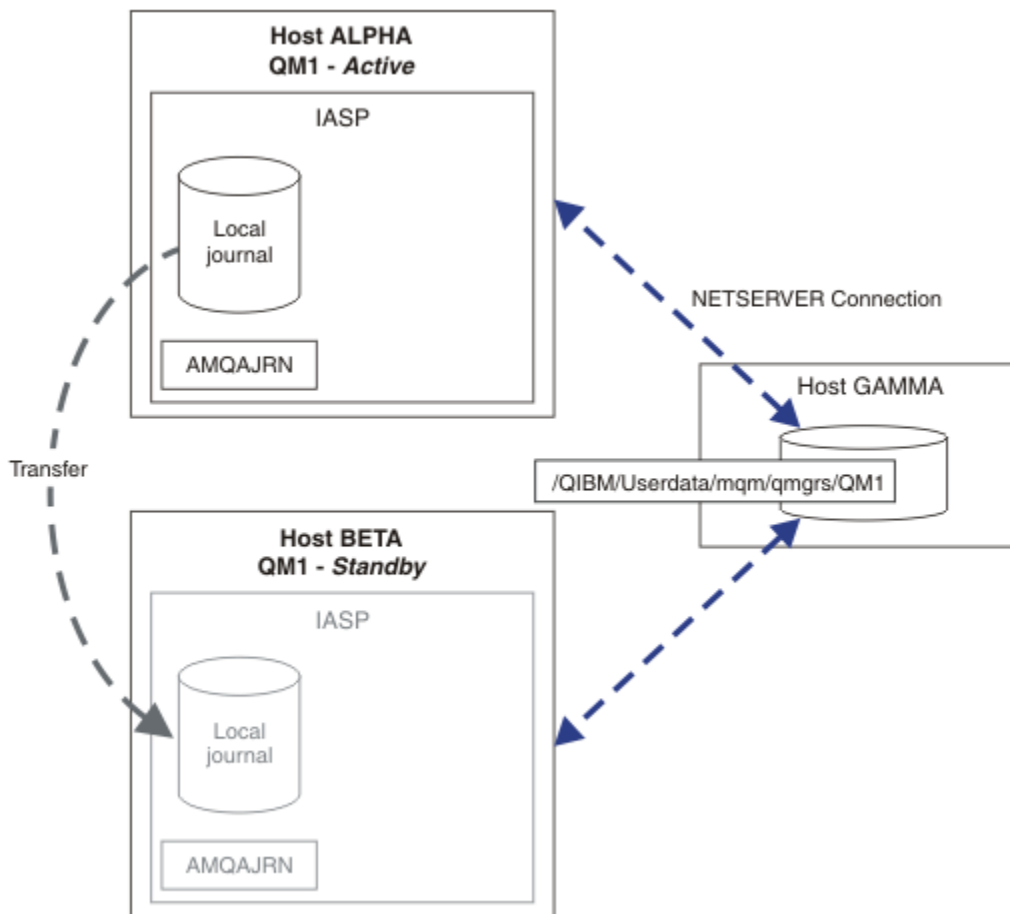


Figura 34. Transferir un diario de gestor de colas utilizando la ASP independiente

## Qué hacer a continuación

- Verifique que las instancias activa y en espera se conmutan automáticamente. Puede ejecutar los programas de alta disponibilidad de muestra para probar la conmutación; consulte el tema [Programas de ejemplo de alta disponibilidad](#). Los programas de ejemplo son clientes 'C'. Puede ejecutarlos desde una plataforma Windows o Unix.
  1. Inicie los programas de ejemplo de alta disponibilidad.
  2. En ALPHA, finalice el gestor de colas solicitando una conmutación:

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) ALSWITCH(*YES)
```

3. Compruebe que la instancia de en BETA esté activa.

#### 4. Reiniciar en ALPHA

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

- Busque configuraciones de alta disponibilidad alternativas:
  1. Utilice NetServer para colocar los datos del gestor de colas en un servidor IBM i.
  2. En lugar de utilizar una ASP independiente para transferir el diario del gestor de colas al servidor en espera, utilice el registro por diario remoto para duplicar el diario en el servidor en espera.

#### **IBM i** *ASP's independientes y alta disponibilidad*

Las ASP independientes permiten mover aplicaciones y datos entre servidores. La flexibilidad de las ASP independientes las ha convertido en la base de algunas soluciones de alta disponibilidad de IBM i. Cuando se plantee la posibilidad de utilizar una ASP o una ASP independiente para el diario de gestor de colas, debería tener en cuenta otra configuración de alta disponibilidad basada en ASP independientes.

Las agrupaciones de almacenamiento auxiliar (ASP) son un bloque de creación de la arquitectura de IBM i. Las unidades de disco se agrupan para formar una única ASP. Al colocar objetos en distintas ASP se puede evitar que los datos de una ASP se infecten debido a las anomalías en el disco de la otra ASP.

Cada servidor IBM i tiene al menos una ASP *básica*, conocida como la agrupación de almacenamiento auxiliar del sistema. Ésta se denomina ASP1 y, a veces se la conoce como \*SYSBAS. Puede configurar hasta 31 ASP de *usuario* básicas adicionales que sean fáciles de distinguir en el sistema de la ASP desde el punto de vista de la aplicación, porque comparten el mismo espacio de nombres. Si utiliza varias ASP básicas para distribuir aplicaciones por muchos discos, puede mejorar el rendimiento y reducir el tiempo de recuperación. La utilización de varias ASP básicas también proporciona un cierto grado de aislamiento contra una posible anomalía del disco, pero no mejora la fiabilidad general.

Las ASP's independientes son un tipo de ASP especial. A menudo se conocen como agrupaciones de discos independientes. Las agrupaciones de discos independientes son un componente clave de la alta disponibilidad de IBM i. Puede almacenar datos y aplicaciones que se consideren a sí mismos como independientes del sistema actual con el que están conectados en unidades de almacenamiento de disco independientes. Puede configurar ASP's independientes conmutables o no conmutables. Desde la perspectiva de la disponibilidad, normalmente sólo tratará con ASP independientes conmutables, que se pueden transferir automáticamente de servidor a servidor. Como consecuencia, podrá mover las aplicaciones y los datos de la ASP independiente de servidor a servidor.

Al contrario que las ASP de usuario básicas, las ASP's independientes no comparten el mismo espacio de nombres que la ASP de sistema. Las aplicaciones que funcionan con ASP de usuario necesitan cambios para que puedan funcionar también con una ASP independiente. Debe verificar si su software y el software de terceros que está utilizando funciona en un entorno de ASP independiente.

Cuando la ASP independiente se adjunta a otro servidor distinto, el espacio de nombres de la ASP independiente tiene que combinarse con el espacio de nombres de la ASP de sistema. Este proceso se denomina *habilitación* de la ASP independiente. Puede habilitar una ASP independiente sin hacer una carga de programa inicial en el servidor. El soporte de agrupación en clúster es necesario para transferir ASP's independientes automáticamente de un servidor al otro.

### **Creación de soluciones fiables con ASP's independientes**

El registro por diario en una ASP independiente, en lugar del registro por diario en una ASP y la utilización de la réplica de diarios constituye un medio alternativo para que el gestor de colas en espera consiga una copia del diario local a partir de la instancia del gestor de colas fallida. Para transferir automáticamente la ASP independiente a otro servidor, deberá haber instalado y configurado el soporte de agrupación en clúster. Existen varias soluciones de alta disponibilidad para ASP's independientes que se basan en el soporte de clústeres y en el duplicado de disco de nivel bajo, que pueden combinarse o sustituirse utilizando gestores de colas multiinstancia.

La siguiente lista describe los componentes que resultan necesarios para crear una solución fiable basada en ASP's independientes.



## **Registro por diario**

Los gestores de colas y otras aplicaciones utilizan diarios para escribir datos persistentes de forma segura en el disco para protegerlos ante posibles pérdidas de datos en la memoria debidas a una anomalía en el servidor. Esto a veces se denomina la coherencia de momentos puntuales. No garantiza la coherencia de las múltiples actualizaciones que tienen lugar durante un período de tiempo.

## **Control de compromiso**

El uso de las transacciones globales permite coordinar las actualizaciones de mensajes y bases de datos a fin de que los datos escritos en el diario sean coherentes. Otorga consistencia a lo largo del tiempo mediante el uso de un protocolo de compromiso de dos fases.

## **Disco conmutado**

Los discos conmutados son gestionados por el grupo de recursos de clúster de dispositivo (CRG) en un clúster de alta disponibilidad. El grupo de recursos de clúster conmuta las ASPs independientes automáticamente en un nuevo servidor cuando se produce una parada no planificada. Los grupos de recursos de clúster están limitados geográficamente en la medida del bus de entrada/salida local.

La configuración de su diario local en una ASPs independiente conmutable permite transferir el diario a un servidor diferente y reanudar el procesamiento de mensajes. Todos los cambios efectuados en mensajes persistentes sin un control de punto de sincronismo o sin la confirmación del control de punto de sincronismo, causan la pérdida de los mismos a menos que la ASP independiente sufra una anomalía.

Si utiliza tanto el control registro por diario y de compromiso en ASPs independientes conmutables, puede transferir los diarios de base de datos y los diarios de gestor de colas a un servidor distinto y reanudar el procesamiento de transacciones sin ninguna pérdida de consistencia ni de transacciones confirmadas.

## **Duplicación entre sistemas (XSM)**

XSM duplica la ASP independiente primaria en una ASP independiente secundaria geográficamente remota por toda la red TCP/IP y transfiere el control automáticamente en caso de que se produzca una anomalía. Tiene la posibilidad de configurar una duplicación síncrona o asíncrona. La duplicación síncrona reduce el rendimiento del gestor de colas porque los datos se duplican antes de que finalicen la operaciones de escritura en el sistema de producción, pero garantizar que la ASP independiente secundaria esté actualizada. Si utiliza la duplicación asíncrona no puede garantizar que la ASP independiente secundaria esté actualizada. La duplicación asíncrona no conserva la coherencia de la ASP independiente secundaria.

Existen tres tecnologías XSM.

### **Duplicación geográfica**

La duplicación geográfica es una ampliación de la agrupación en clúster que permite conmutar las ASPs independientes en una amplia zona. Tanto en modalidad síncrona como asíncrona. Sólo se puede garantizar una alta disponibilidad en modalidad síncrona, pero la separación de las ASPs independientes podría afectar demasiado al rendimiento. Puede combinar con la duplicación geográfica con el disco conmutado para proporcionar una alta disponibilidad local y una recuperación tras desastre remota.

### **Duplicación síncrona de distancia corta**

La duplicación síncrona de distancia corta es un servicio de nivel de dispositivo que proporciona una duplicación síncrona local rápida que cubre distancias más largas que el bus local. Puede combinarla con un gestor de colas multiinstancia para obtener una alta disponibilidad del gestor de colas y, al tener dos copias de la ASP independiente, obtener una alta disponibilidad del diario del gestor de colas.

### **Duplicación síncrona de distancia larga**

La duplicación síncrona de distancia larga es un servicio de nivel de dispositivo que proporciona duplicación asíncrona y resulta adecuada para hacer copias de seguridad y conseguir recuperaciones tras desastre en distancias más largas, pero no resulta una opción habitual para la alta disponibilidad, porque sólo conserva la coherencia de un momento puntual en lugar de una vigencia prolongada.

Los puntos más importantes que debe tener en cuenta para tomar una decisión son,

### ¿ASP o ASP independiente?

No es necesario que ejecute un clúster de alta disponibilidad de IBM i para utilizar gestores de colas multiinstancia. Puede elegir ASP independientes, si ya está utilizando ASP independientes o si tiene requisitos de disponibilidad para otras aplicaciones que requieran ASP independientes. Podría valer la pena combinar las ASP independientes con gestores de colas multiinstancia para sustituir la supervisión del gestor de colas como una forma para detectar una posible anomalía del gestor de colas.

### ¿Disponibilidad?

¿Cuál es el objetivo de tiempo de recuperación (RTO)? Si necesita la apariencia de un comportamiento casi ininterrumpido... ¿qué solución tiene el tiempo de recuperación más rápido?

### ¿Disponibilidad de diario?

Cómo se elimina el diario como un único punto de error. Podrá adoptar una solución de hardware utilizando dispositivos RAID 1 o mejores, o podría combinar o utilizar una solución de software utilizando la réplica de diarios o el duplicado de discos.

### ¿Distancia?

Qué distancia separa las instancias del gestor de colas activa y en espera. ¿Sus usuarios pueden tolerar la degradación del rendimiento al hacer réplicas síncronas por distancias superiores a unos 250 metros?

### ¿Habilidades?

Es necesario trabajar para automatizar las tareas administrativas implicadas en el mantenimiento y el ejercicio de la solución de forma regular. Las habilidades necesarias para realizar la automatización son diferentes para las soluciones basadas en ASPs y en ASPs independientes.

**IBM i**

## **Borrado de un gestor de colas multiinstancia en IBM i**

Antes de suprimir un gestor de colas multiinstancia, detenga el registro por diario remoto y elimine las instancias del gestor de colas.

### Antes de empezar

1. En este ejemplo, hay dos instancias del gestor de colas QM1 definidas en los servidores ALPHA y BETA. ALPHA es la instancia activa y BETA está en espera. Los datos de gestor de colas asociados al gestor de colas QM1 se almacenan en el servidor IBM i GAMMA, utilizando NetServer. Consulte “Creación de un gestor de colas multiinstancia mediante la duplicación de diarios y NetServer en IBM i” en la [página 411](#).
2. ALPHA y BETA deben estar conectados para que IBM MQ pueda suprimir todos los diarios remotos definidos.
3. Verifique que sea posible acceder al directorio /QNTC y a la compartición de archivos de directorio de servidor mediante los mandatos del sistema **EDTF** o **WRKLNK**

### Acercas de esta tarea

Antes de suprimir un gestor de colas multiinstancia de un servidor mediante el mandato **DLTMQM**, elimine todas las instancias del gestor de colas de otros servidores mediante el mandato **RMVMQMINF**.

Cuando elimine una instancia de gestor de colas mediante el mandato **RMVMQMINF**, se suprimirán los diarios locales y remotos cuyo prefijo sea AMQ, y estén asociados a la instancia. También se suprimirá la información de configuración sobre la instancia de gestor de colas, local respecto al servidor.

No ejecute el mandato **RMVMQMINF** en el servidor que contenga la instancia restante del gestor de colas. Al hacerlo, se impide que **DLTMQM** funcione correctamente.

Suprima el gestor de colas mediante el mandato **DLTMQM**. Los datos del gestor de colas se eliminan del compartimiento de red. Se suprimen los diarios locales y remotos cuyo prefijo sea AMQ y estén asociados a la instancia. **DLTMQM** también suprime la información de configuración de la instancia del gestor de colas, local respecto al servidor.

En el ejemplo sólo hay dos instancias del gestor de colas. IBM MQ da soporte a una configuración multiinstancia en ejecución que tenga una instancia de gestor de colas activa y una instancia en espera. Si ha creado instancias de gestor de colas adicionales para utilizarlas en configuraciones que están en ejecución, elimínelas mediante el mandato **RMVMQMINF**, antes de suprimir la instancia restante.

## Procedimiento

1. Ejecute el mandato **CHGMQMJRN RMTJRNSTS** (\*INACTIVE) en cada servidor para desactivar el registro por diario remoto entre las instancias del gestor de colas.

a) En ALPHA:

```
CHGMQMJRN MQMNAME('QM1')
RMTJRNRDB('BETA') RMTJRNSTS(*INACTIVE)
```

b) En BETA:

```
CHGMQMJRN MQMNAME('QM1')
RMTJRNRDB('ALPHA') RMTJRNSTS(*INACTIVE)
```

2. Ejecute el mandato **ENDMQM** en ALPHA, la instancia de gestor de colas activa, para detener ambas instancias de QM1.

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) INSTANCE(*ALL) ENDCCTJOB(*YES)
```

3. Ejecute el mandato **RMVMQMINF** en ALPHA para eliminar los recursos del gestor de colas de la instancia de ALPHA y BETA.

```
RMVMQMINF MQMNAME(QM1)
```

**RMVMQMINF** elimina, de ALPHA, la información de configuración del gestor de colas para QM1. Si el nombre de diario tiene el prefijo AMQ, se suprime, de ALPHA, el diario local asociado a QM1. Si el nombre de diario tiene el prefijo AMQ y se ha creado un diario remoto, también se elimina, de BETA, el diario remoto.

4. Ejecute el mandato **DLTMQM** en BETA para suprimir QM1.

```
DLTMQM MQMNAME(QM1)
```

**DLTMQM** suprime los datos del gestor de colas de la unidad compartida de red en GAMMA. Elimina, de BETA, la información de configuración del gestor de colas para QM1. Si el nombre de diario tiene el prefijo AMQ, se suprime, de BETA, el diario local asociado a QM1. Si el nombre de diario tiene el prefijo AMQ y se ha creado un diario remoto, también se elimina, de ALPHA, el diario remoto.

## Resultados

**DLTMQM** y **RMVMQMINF** suprimen los diarios locales y remotos que han creado **CRTMQM** y **ADDMQJRN**. Los mandatos también suprimen los destinatarios de diario. Los diarios y los receptores de diario deben seguir el convenio de denominación, que implica tener nombres que empiecen por AMQ. **DLTMQM** y **RMVMQMINF** eliminan los objetos de gestor de colas, los datos del gestor de colas, y la información de configuración del gestor de colas del archivo `mq5.ini`.

## Qué hacer a continuación

Un enfoque alternativo es emitir los mandatos siguientes después de desactivar el registro por diario en el paso “1” en la [página 427](#), y antes de finalizar las instancias del gestor de colas. O, si no ha seguido el convenio de denominación, debe suprimir los diarios y los receptores de diario por nombre.

1. En ALPHA:

```
RMVMQMJRN MQMNAME('QM1') RMTJRNRDB('BETA')
```

2. En BETA:

```
RMVMQMJRN MQMNAME('QM1') RMTJRNRDB('ALPHA')
```

Después de suprimir los diarios, prosiga con el resto de los pasos.

## **IBM i** *Copia de seguridad de un gestor de colas multiinstancia en IBM i*

En el procedimiento se muestra cómo realizar una copia de seguridad de los objetos de gestor de colas en el servidor local y de los datos del gestor de colas en el archivo del servidor de red. Adapte el ejemplo para realizar una copia de seguridad de datos para otros gestores de colas.

### **Antes de empezar**

En este ejemplo, los datos de gestor de colas asociados al gestor de colas QM1 se almacenan en el servidor IBM i llamado GAMMA, utilizando NetServer. Consulte el apartado [“Creación de un gestor de colas multiinstancia mediante la duplicación de diarios y NetServer en IBM i”](#) en la página 411. IBM MQ está instalado en los servidores ALPHA y BETA. El gestor de colas, QM1, está configurado en ALPHA y BETA.

### **Acerca de esta tarea**

IBM i no permite guardar datos desde un directorio remoto. Guarde los datos del gestor de colas en un sistema de archivos remoto utilizando los procedimientos de copia de seguridad local en el servidor del sistema de archivos. En esta tarea, el sistema de archivos de red está en un servidor IBM i, GAMMA. Se realiza una copia de seguridad de los datos del gestor de colas en un archivo de salvar en GAMMA.

Si el sistema de archivos de red estuviera en Windows o Linux, podría almacenar los datos del gestor de colas en un archivo comprimido y luego guardarlos. Si tiene un sistema de copia de seguridad como, por ejemplo, Tivoli Storage Manager, utilícelo para realizar la copia de seguridad de los datos del gestor de colas.

### **Procedimiento**

1. Cree un archivo de salvar en ALPHA para la biblioteca del gestor de colas asociada a QM1.

Utilice el nombre de la biblioteca del gestor de colas para denominar el archivo de salvar.

```
CRTSAVF FILE(QGPL/QMQM1)
```

2. Guarde la biblioteca del gestor de colas en el archivo de salvar en ALPHA.

```
SAVLIB LIB(QMQM1) DEV(*SAVF) SAVF(QGPL/QMQM1)
```

3. Cree un archivo de salvar para el directorio de datos del gestor de colas, en GAMMA.

Utilice el nombre del gestor de colas para denominar el archivo de salvar.

```
CRTSAVF FILE(QGPL/QMDQM1)
```

4. Guarde la copia de los datos del gestor de colas del directorio local en GAMMA.

```
SAV DEV('/QSYS.LIB/QGPL.LIB/QMDQM1.FILE') OBJ('/QIBM/Userdata/mqm/qmgrs/QM1')
```

## Mandatos para configurar gestores de colas multiinstancia

IBM MQ tiene mandatos para simplificar las tareas de configurar la réplica de diarios, añadir nuevas instancias de gestor de colas y configurar gestores de colas para utilizar la ASP independiente.

Los mandatos del diario para crear y gestionar diarios locales y remotos son,

### ADDMQMJRN

Con este mandato puede crear diarios locales y remotos con nombres para una instancia del gestor de colas y configurar si la réplica es síncrona o asíncrona, cual es el tiempo de espera de la síncrona y si el diario remoto debe activarse inmediatamente.

### CHGMQMJRN

El mandato modifica el tiempo de espera, el estado y los parámetros de entrega que afectan a los diarios duplicados.

### RMVMQMJRN

Elimina los diarios *remotos* con nombre de una instancia del gestor de colas.

### WRKMQMJRN

Lista el estado de los diarios locales y remotos para una instancia de gestor de colas local.

Añada y gestione instancias del gestor de colas adicionales utilizando los siguientes mandatos, que modifican el archivo `mqs.ini`.

### ADDMQMINF

Este mandato utiliza información extraída del archivo `mqs.ini` con el mandato `DSPMQMINF` para añadir una nueva instancia del gestor de colas en un servidor IBM i distinto.

### RMVMQMINF

Elimina una instancia del gestor de colas. Utilice este mandato, o bien para eliminar una instancia de un gestor de colas existente, o bien para eliminar la información de configuración de un gestor de colas que se ha suprimido de un servidor distinto.

El mandato **CRTMQM** tiene tres parámetros que le ayudarán a configurar un gestor de colas multiinstancia.

### MQMDIRP(\*DFT | *prefijo-directorio*)

Utilice este parámetro para seleccionar un punto de montaje que se correlacione con los datos del gestor de colas en almacenamiento por red.

### ASP(\*SYSTEM|\*ASPDEV|*número-agrupación-almacenamiento-auxiliar*)

Especifique `*SYSTEM` o un *número-agrupación-almacenamiento-auxiliar* para colocar el diario del gestor de colas en el sistema o en una ASP de usuario básica. Seleccione la opción `*ASPDEV` y establezca también un nombre de dispositivo utilizando el parámetro **ASPDEV** para colocar el diario del gestor de colas en una ASP independiente.

### ASPDEV(\*ASP|*nombre-dispositivo*)

Especifique un *nombre-dispositivo* de un dispositivo de ASP independiente primario o secundario. Si selecciona `*ASP` obtendrá el mismo resultado que si especifica **ASP** (`*SYSTEM`).

## Consideraciones sobre la migración tras error de disco y el rendimiento en IBM i

Utilice distintas agrupaciones de almacenamiento auxiliares para mejorar el rendimiento y la fiabilidad.

Si utiliza un número elevado de mensajes persistentes o mensajes de gran tamaño en sus aplicaciones, el tiempo empleado en guardar dichos mensajes en el disco pasará a ser un factor fundamental en el rendimiento del sistema.

Asegúrese de que tiene suficiente activación de disco para hacer frente a esta posibilidad; también podría tener una "Auxiliary Storage Pool" (agrupación de almacenamiento auxiliar - ASP) aparte en la que guardar los receptores de diario del gestor de colas.

Puede especificar en qué ASP se almacena la biblioteca de gestores de cola y los diarios cuando crea el gestor de colas mediante el parámetro `ASP` de **CRTMQM**. Por omisión, los diarios y la biblioteca de gestores de colas así como los datos IFS se almacenan en el sistema ASP.

Las ASP permiten aislar los objetos en una o varias unidades de disco específicas. Con ello, también se reduce la pérdida de datos debido a una anomalía en el soporte de disco. En la mayoría de los casos, sólo se pierden los datos almacenados en unidades de disco de la ASP afectada.

Se le recomienda almacenar la biblioteca de gestores de colas y los datos de diario en ASP de usuario separadas a fin de que la raíz del sistema de archivos IFS proporcione sustitución por anomalía y reduzca la contención de disco.

Para obtener más información, consulte [Copia de seguridad y recuperación](#) en la documentación de IBM i.

## IBM i Utilización de SAVLIB para guardar bibliotecas de IBM MQ en IBM i

No puede utilizar SAVLIB LIB(\*ALLUSR) para guardar las bibliotecas de IBM MQ porque estas bibliotecas tiene nombres que empiezan por Q.

Puede utilizar SAVLIB LIB(QM\*) para guardar todas las bibliotecas del gestor de colas, pero sólo si está utilizando un dispositivo de guardar distinto de \*SAVF. Para DEV(\*SAVF), debe utilizar un mandato SAVLIB para todas y cada una de las bibliotecas de gestor de colas del sistema.

## IBM i Progresivo IBM MQ for IBM i

En esta sección se explica cómo desactivar temporalmente (finalizar ordenadamente) IBM MQ for IBM i.

Para desactivar temporalmente IBM MQ for IBM i:

1. Inicie una nueva sesión interactiva de IBM MQ for IBM i, asegurándose de no acceder a ningún objeto.
2. Asegúrese de que tiene:
  - Autorización \*ALLOBJ o autorización de gestión de objetos para la biblioteca QMQM
  - Autorización suficiente para utilizar el mandato ENDSBS
3. Advierta a todos los usuarios de que va a detener IBM MQ for IBM i.
4. Los pasos que debe seguir a continuación dependen de si desea concluir (desactivar temporalmente) un solo gestor de colas (mientras que otros pueden seguir existiendo) (consulte [“Conclusión de un gestor de colas individual para IBM MQ for IBM i”](#) en la página 431) o todos los gestores de colas (consulte [“Conclusión de todos los gestores de colas para IBM MQ for IBM i”](#) en la página 432).
5. Concluya el servidor mqweb especificando el mandato siguiente en qshell:

```
/QIBM/ProdData/mqm/bin/endmqweb
```

### Parámetro de ENDMQM ENDCCTJOB(\*YES)

El parámetro ENDCCTJOB(\*YES) de ENDMQM funciona de forma diferente en IBM MQ for IBM i V6.0 y posteriores en comparación con las versiones anteriores.

En las versiones anteriores, cuando se especifica ENDCCTJOB(\*YES), MQ finaliza las aplicaciones por la fuerza.

En IBM MQ for IBM i V6.0 o posteriores, cuando se especifica ENDCCTJOB(\*YES), las aplicaciones no finalizan sino que se desconectan del gestor de colas.

Si especifica ENDCCTJOB(\*YES) y tiene aplicaciones que no están escritas para detectar que un gestor de colas está finalizando, la próxima vez que se emita una nueva llamada MQI, la llamada devolverá un error MQRC\_CONNECTION\_BROKEN (2009).

Como alternativa al uso de ENDCCTJOB(\*YES), utilice el parámetro ENDCCTJOB(\*NO) y la opción 22 de WRKMQM (Trabajar con trabajos) para finalizar manualmente los trabajos de aplicaciones que impedirán que un gestor de colas se reinicie.

Utilice esta información para conocer los tres tipos de conclusión.

En los procedimientos que se describen a continuación, se utiliza el nombre de gestor de colas de ejemplo QMgr1 y el nombre de subsistema de ejemplo SUBX. Sustituya estos nombres por sus propios valores si es necesario.

### Conclusión planificada

Conclusión planificada de un gestor de colas en IBM i

1. Antes de cerrar, ejecute:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(QMgr1) DSPJRNDTA(*YES)
```

2. Para concluir el gestor de colas, ejecute:

```
ENDMQM MQMNAME(QMgr1) OPTION(*CNTRLD)
```

Si QMgr1 no finaliza, el canal o las aplicaciones probablemente están ocupados.

3. Si debe concluir QMgr1 inmediatamente, ejecute lo siguiente:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)  
ENDCCTJOB(*YES) TIMEOUT(15)
```

### Conclusión no planificada

1. Para concluir el gestor de colas, ejecute:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)
```

Si QMgr1 no finaliza, el canal o las aplicaciones probablemente están ocupados.

2. Si necesita concluir QMgr1 inmediatamente, ejecute lo siguiente:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)  
ENDCCTJOB(*YES) TIMEOUT(15)
```

### Cierre en condiciones anómalas

1. Para concluir el gestor de colas, ejecute:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)
```

Si QMgr1 no finaliza, continúe con el paso 3 siempre que:

- QMgr1 se encuentre en su propio subsistema, o bien
  - Pueda finalizar todos los gestores de colas que comparten el mismo subsistema que QMgr1. Utilice el procedimiento de conclusión no planificada para todos los gestores de colas de este tipo.
2. Cuando haya llevado a cabo todos los pasos del procedimiento para todos los gestores de colas que comparten el subsistema (SUBX en nuestros ejemplos), ejecute:

```
ENDSBS SUBX *IMMED
```

Si este mandato no se lleva a cabo correctamente, concluya todos los gestores de colas, utilizando el procedimiento de conclusión no planificada, y realice una IPL en la máquina.

**Aviso:** No utilice ENDJOBABN para trabajos de IBM MQ que no finalicen como resultado de ENDJOB o ENDSBS, a menos que esté preparado para realizar una IPL en la máquina inmediatamente después.

3. Inicie el subsistema, ejecutando:

```
STRSBS SUBX
```

4. Concluya el gestor de colas inmediatamente, ejecutando:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)  
ENDCCTJOB(*YES) TIMEOUT(10)
```

5. Reinicie el gestor de colas, ejecutando:

```
STRMQM MQMNAME(QMgr1)
```

Si esto falla, y:

- Ha reiniciado la máquina realizando una IPL, o bien
- Tiene un solo gestor de colas

Reorganice la memoria compartida de IBM MQ ejecutando:

```
ENDMQM MQMNAME(*ALL) OPTION(*IMMED)  
ENDCCTJOB(*YES) TIMEOUT(15)
```

antes de repetir el paso 5.

Si el gestor de colas tarda más de unos cuantos segundos en reiniciarse, IBM MQ añade mensajes de estado de forma intermitente en el registro de trabajo, detallando el progreso del inicio.

Si sigue teniendo problemas para reiniciar el gestor de colas, póngase en contacto con el servicio de soporte de IBM. Cualquier otra acción que pudiera realizar podría dañar el gestor de colas, dejando a IBM MQ en un estado imposible de recuperar.

**IBM i**

## Conclusión de todos los gestores de colas para IBM MQ for IBM i

Utilice esta información para conocer los tres tipos de conclusión.

Los procedimientos son prácticamente los mismos que para un solo gestor de colas, pero utilizando \*ALL en lugar del nombre del gestor de colas cuando sea posible, o utilizando un mandato repetidamente con cada nombre de gestor de colas correspondiente. En todos los procedimientos, se utiliza el nombre de gestor de colas de ejemplo QMgr1 y el nombre de subsistema de ejemplo SUBX. Puede sustituir estos nombres por otros que desee.

### Conclusión planificada

1. Una hora antes de la conclusión, ejecute:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(QMgr1) DSPJRNDTA(*YES)
```

Repita este paso para cada gestor de colas que desee concluir.

2. Para concluir el gestor de colas, ejecute:

```
ENDMQM MQMNAME(QMgr1) OPTION(*CNTRLD)
```

Repita este paso para cada gestor de colas que desee concluir; los mandatos separados pueden ejecutarse en paralelo.



Si algún gestor de colas no finaliza al cabo de un período de tiempo razonable (por ejemplo, 10 minutos), continúe en el paso 3.

3. Para concluir todos los gestores de colas inmediatamente, ejecute lo siguiente:

```
ENDMQM MQMNAME(*ALL) OPTION(*IMMED)  
ENDCCTJOB(*YES) TIMEOUT(15)
```

## Conclusión no planificada

1. Para concluir un gestor de colas, ejecute:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)
```

Repita este paso para cada gestor de colas que desee concluir; los mandatos separados pueden ejecutarse en paralelo.

Si los gestores de colas no finalizan, el canal o las aplicaciones probablemente están ocupados.

2. Si necesita concluir los gestores de colas inmediatamente, ejecute lo siguiente:

```
ENDMQM MQMNAME(*ALL) OPTION(*IMMED)  
ENDCCTJOB(*YES) TIMEOUT(15)
```

## Cierre en condiciones anómalas

1. Para concluir los gestores de colas, ejecute:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)
```

Repita este paso para cada gestor de colas que desee concluir; los mandatos separados pueden ejecutarse en paralelo.

2. Finalice los subsistemas (SUBX en nuestros ejemplos), ejecutando:

```
ENDSBS SUBX *IMMED
```

Repita este paso para cada subsistema que desee concluir; los mandatos separados pueden ejecutarse en paralelo.

Si este mandato no se ejecuta correctamente, realice una IPL en el sistema.

**Aviso:** No utilice ENDJOBABN para trabajos que no finalicen como resultado de ENDJOB o ENDSBS, a menos que esté preparado para realizar una IPL en el sistema inmediatamente después.

3. Inicie los subsistemas, ejecutando:

```
STRSBS SUBX
```

Repita este paso para cada subsistema que desee iniciar.

4. Concluya los gestores de colas inmediatamente, ejecutando:

```
ENDMQM MQMNAME(*ALL) OPTION(*IMMED)  
ENDCCTJOB(*YES) TIMEOUT(15)
```

5. Reinicie los gestores de colas, ejecutando:

```
STRMQM MQMNAME(QMgr1)
```

Repita este paso para cada gestor de colas que desee iniciar.

Si algún gestor de colas tarda más de unos cuantos segundos en reiniciarse, IBM MQ mostrará mensajes de estado de forma intermitente que detallan el progreso del inicio.

Si sigue teniendo problemas para reiniciar algún gestor de colas, póngase en contacto con el servicio de soporte de IBM. Cualquier otra acción que pudiera realizar podría dañar los gestores de colas, dejando a MQSeries o IBM MQ en un estado imposible de recuperar.

z/OS

## Administración de IBM MQ for z/OS

La administración de gestores de colas y recursos asociados incluye las tareas que se realizan con frecuencia para activar y gestionar estos recursos. Elija el método que prefiera para administrar los gestores de colas y los recursos asociados.

IBM MQ for z/OS puede ser controlado y gestionado por un conjunto de programas de utilidad proporcionados con el producto. Puede utilizar los mandatos de script IBM MQ (MQSC) o los formatos de mandato programable (PCF) para administrar IBM MQ for z/OS. Para obtener información sobre el uso de mandatos en IBM MQ for z/OS, consulte [“Emisión de mandatos a IBM MQ for z/OS”](#) en la página 434.

IBM MQ for z/OS también proporciona un conjunto de programas de utilidad para ayudarle con la administración del sistema. Para obtener información sobre los distintos programas de utilidades y cómo utilizarlos, consulte [“Los programas de utilidad de IBM MQ for z/OS”](#) en la página 443.

Si desea más detalles sobre cómo administrar IBM MQ for z/OS y las distintas tareas administrativas que podría tener que llevar a cabo, consulte los enlaces siguientes:

### Conceptos relacionados

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

### Tareas relacionadas

[“Trabajar con objetos de IBM MQ locales”](#) en la página 142

Se pueden administrar objetos locales de IBM MQ para soportar programas de aplicación que usen la interfaz de cola de mensajes (Message Queue Interface, MQI).

[“Trabajar con objetos de IBM MQ remotos”](#) en la página 207

Puede administrar objetos IBM MQ en gestor de colas remotos utilizando mandatos MQSC, mandatos PCF o la administrative REST API. Antes de poder utilizar cualquiera de estos métodos, debe definir colas de transmisión y canales entre el gestor de colas local y el gestor de colas remoto, para que los mandatos se puedan enviar al gestor de colas remoto y las respuestas recibidas por el gestor de colas local. De forma alternativa, puede configurar un clúster de gestor de colas y, después, utilizar los mismos métodos de administración remota.

[“Administración de IBM MQ”](#) en la página 7

Para administrar los gestores de colas de IBM MQ y los recursos asociados, elija el método que prefiera de un conjunto de tareas que puede utilizar para activar y gestionar estos recursos.

### Planificación

[Planificación del entorno de IBM MQ en z/OS](#)

### Configuración

[Configuración de z/OS](#)

[Utilización de los programas de utilidad de IBM MQ for z/OS](#)

### Referencia relacionada

[“Orígenes desde los que puede emitir mandatos MQSC en z/OS”](#) en la página 20

Los mandatos MQSC se pueden emitir desde varios orígenes, en función del mandato.

[Referencia de formatos de mandato programable](#)

z/OS

## Emisión de mandatos a IBM MQ for z/OS

Puede utilizar mandatos de script de IBM MQ (MQSC) en modalidad interactiva o por lotes para controlar un gestor de colas.

IBM MQ for z/OS da soporte a mandatos MQSC, que se pueden emitir desde los siguientes orígenes:

- La consola de z/OS o equivalente (como SDSF/TSO).
- Los conjuntos de datos de entrada de inicialización.
- El programa de utilidad por lotes proporcionado, CSQUTIL, que procesa una lista de mandatos en un conjunto de datos secuencial.
- Una aplicación debidamente autorizada, enviando un mandato como un mensaje a la cola de entrada de mandatos. La aplicación puede ser cualquiera de las siguientes:
  - Un programa de región por lotes
  - Una aplicación CICS
  - Una aplicación de IMS
  - Una aplicación TSO
  - Un programa de aplicación o programa de utilidad en otro sistema IBM MQ

En la [Tabla 31 en la página 438](#) se resumen los mandatos MQSC y los orígenes desde los que se pueden emitir.

Gran parte de la funcionalidad de estos mandatos está disponible de forma conveniente en los paneles de operaciones y los paneles de control de IBM MQ for z/OS.

Los cambios realizados en las definiciones de recurso de un gestor de colas mediante los mandatos (directa o indirectamente) se conservan después de los reinicios del subsistema de IBM MQ.

IBM MQ for z/OS también da soporte a los mandatos PCF (formato de mandato programable). Estos mandatos simplifican la creación de aplicaciones para la administración de IBM MQ. Los mandatos MQSC tienen un formato de texto legible, mientras que el formato de mandato programable (PCF) permite a las aplicaciones crear solicitudes y leer las respuestas sin tener que analizar series de texto. Al igual que los mandatos MQSC, las aplicaciones emiten mandatos PCF enviándolos como mensajes a la cola de entrada de mandatos. Para obtener más información sobre cómo utilizar mandatos PCF y para obtener detalles sobre los mandatos, consulte la documentación [Referencia de formatos de mandato programable](#).

## **Definiciones privadas y globales en IBM MQ for z/OS**

Al definir un objeto en IBM MQ for z/OS, puede elegir si desea compartir la definición con otros gestores de colas (una definición *global*) o si la definición de objeto será utilizada por un solo gestor de colas (una definición *privada*). Esto se denomina el objeto *disposición*.

### **Definición global**

Si el gestor de colas pertenece a un grupo de compartición de colas, puede decidir compartir todas las definiciones de objeto con los otros miembros del grupo. Esto significa que debe definir un objeto una única vez, reduciendo el número total de definiciones necesarias para todo el sistema.

Las definiciones de objeto globales se mantienen en un *repositorio compartido* (una base de datos compartida de Db2) y están disponibles para todos los gestores de colas del grupo de compartición de colas. Estos objetos tienen una disposición de GROUP.

### **Definición privada**

Si desea crear una definición de objeto que sólo sea necesaria para un gestor de colas, o si el gestor de colas no es miembro de un grupo de compartición de colas, puede crear definiciones de objeto que no se compartan con otros miembros de un grupo de compartición de colas.

Las definiciones de objeto privado se conservan en el conjunto de páginas cero del gestor de colas que realiza la definición. Estos objetos tienen una disposición de QMGR.

Puede crear definiciones privadas para todos los tipos de objetos de IBM MQ, excepto para estructuras de recurso de acoplamiento (es decir, canales, listas de nombres, definiciones de proceso, colas, gestores de colas, definiciones de clase de almacenamiento y objetos de información de autenticación) y para definiciones globales para todos los tipos de objetos excepto gestores de colas.

IBM MQ copia automáticamente la definición de un objeto de grupo en el conjunto de páginas cero de cada gestor de colas que lo utiliza. Puede modificar la copia de la definición temporalmente si lo desea, y IBM MQ le permite renovar las copias del conjunto de páginas desde la copia del repositorio en caso necesario.

IBM MQ siempre trata de renovar las copias del conjunto de páginas a partir de la copia del repositorio durante el inicio (en los mandatos de canal, esto se hace al reiniciar iniciador de canal) o si se modifica el objeto de grupo.

**Nota:** La copia de la definición se renueva desde la definición del grupo, solo si la definición del grupo ha cambiado desde que se ha creado la copia de la definición.

Esto garantiza que las copias del conjunto de páginas reflejen la versión en el repositorio, incluidos todos los cambios que se hayan realizado cuando el gestor de colas estaba inactivo. Las copias se renuevan generando mandatos DEFINE REPLACE, por ello, existen circunstancias bajo las que no se realiza la renovación, por ejemplo:

- Si hay un copia de la cola abierta, falla la renovación que cambia el uso de la cola.
- Si una copia de una cola contiene mensajes, falla la renovación que suprime dicha cola.
- Si hay una copia de una cola requiriera ALTER con FORCE para cambiarla.

En estas circunstancias, la renovación no se realiza en dicha copia, pero se lleva a cabo en las copias de todos los demás gestores de colas.

Si el gestor de colas concluye y luego se reinicia autónomamente, todas las copias locales de objetos se suprimen, a menos que, por ejemplo, la cola tenga mensajes asociados.

Existe una tercera disposición de objeto que sólo se aplica a las colas locales. Ello permite crear colas compartidas. La definición de una cola compartida se mantiene en el repositorio compartido y está disponible para todos los gestores de colas del grupo de compartición de colas. Además, los mensajes de una cola compartida también están disponibles para todos los gestores de colas del grupo de compartición de colas. Esto se describe en [Grupos de compartición de colas y colas compartidas](#). Las colas compartidas tienen una disposición de objeto de SHARED.

La siguiente tabla resume el efecto de las opciones de disposición de objeto en los gestores de colas iniciados autónomamente y como miembro de un grupo de compartición de colas.

Disposición	Gestor de colas autónomo	Miembro de un grupo de compartición de colas
QMGR	Definición de objeto conservada en el conjunto de páginas cero.	Definición de objeto conservada en el conjunto de páginas cero.
GROUP	No permitido.	Definición de objeto retenida en el repositorio compartido. Copia local conservada en el conjunto de páginas cero de cada gestor de colas del grupo.
SHARED	No permitido.	Definición de cola conservada en el repositorio compartido. Mensajes disponibles en cualquier gestor de colas del grupo.

## Manipulación de definiciones globales

Si desea cambiar la definición de un objeto que se conserva en el repositorio compartido, debe especificar si desea cambiar la versión en el repositorio o la copia local en el conjunto de páginas cero. Utilice la disposición del objeto como parte del mandato para hacerlo.

## **Direccionamiento de mandatos a gestores de colas diferentes en z/OS**

Puede utilizar el *ámbito de mandato* para controlar el gestor de colas en el que se ejecuta el mandato.

Puede optar por ejecutar un mandato en el gestor de colas en el que se introduce o en un gestor de colas diferente del grupo de compartición de colas. También puede determinar emitir un mandato determinado en paralelo en todos los gestores de colas de un grupo de compartición de colas. Se puede hacer tanto para los mandatos MQSC como para los PCF.

Ello viene determinado por el *ámbito de mandato*. El ámbito del mandato se utiliza con la disposición de objeto para determinar con qué versión de un objeto desea trabajar.

Por ejemplo, es posible que desee modificar algunos de los atributos de un objeto, cuya definición se conserve en el repositorio compartido.

- Es posible que desee cambiar la versión pero en un solo gestor de colas, y no hacer cambios a la versión del repositorio o en las que estén siendo utilizadas por otros gestores de colas.
- Es posible que desee cambiar la versión del repositorio compartido para futuros usuarios, pero que quiera dejar las copias existentes sin modificar.
- Es posible que desee cambiar la versión del repositorio compartido, pero también desee que los cambios se reflejen inmediatamente en todos los gestores de colas del grupo de compartición de colas que contienen una copia del objeto en su conjunto de páginas cero.

Utilice el ámbito de mandato para especificar si el mandato debe ejecutarse en este gestor de colas, en otro gestor de colas o en todos los gestores de colas. Utilice la disposición de objeto para especificar si el objeto que está manipulando se encuentra en el repositorio compartido (un objeto de grupo) o es una copia local del el conjunto de páginas cero (un objeto del gestor de colas).

No es necesario que especifique el ámbito de mandato y la disposición de objeto para trabajar con una cola compartida porque cada gestor de colas del grupo de compartición de colas gestiona la cola compartida como una sola cola.

## **Resumen de mandatos en IBM MQ for z/OS**

Utilice este tema como referencia de los mandatos MQSC y PCF principales.

En la [Tabla 30](#) en la [página 437](#) se resumen los mandatos MQSC y PCF que están disponibles en IBM MQ for z/OS para modificar, definir, suprimir y visualizar objetos de IBM MQ.

<i>Tabla 30. Resumen de los mandatos MQSC y PCF principales por tipo de objeto</i>				
<b>Mandato MQSC</b>	<b>ALTER</b>	<b>DEFINE</b>	<b>DISPLAY</b>	<b>DELETE</b>
<b>Mandato PCF</b>	<b>Cambiar</b>	<b>Crear/Copiar</b>	<b>Consultar</b>	<b>Suprimir</b>
AUTHINFO	X	X	X	X
CFSTATUS			X	
CFSTRUCT	X	X	X	X
CHANNEL	X	X	X	X
CHSTATUS			X	
LISTA DE NOMBRES	X	X	X	X
PROCESS	X	X	X	X
QALIAS	M	M	M	M
QCLUSTER			M	

Tabla 30. Resumen de los mandatos MQSC y PCF principales por tipo de objeto (continuación)

Mandato MQSC	ALTER	DEFINE	DISPLAY	DELETE
QLOCAL	M	M	M	M
QMGR	X		X	
QMODEL	M	M	M	M
QREMOTE	M	M	M	M
QUEUE	P	P	X	P
QSTATUS			X	
STGCLASS	X	X	X	X

**Clave para la tabla de símbolos:**

- M = Sólo MQSC
- P = Sólo PCF
- X = Ambos

Hay muchos otros mandatos MQSC y PCF que le permiten gestionar otros recursos de IBM MQ y llevar a cabo otras acciones además de las resumidas en la [Tabla 30](#) en la [página 437](#).

La [Tabla 31](#) en la [página 438](#) muestra cada mandato MQSC y desde dónde puede emitirse cada mandato:

- Conjunto de datos de entrada de inicialización CSQINP1
- Conjunto de datos de entrada de inicialización CSQINP2
- Consola de z/OS (o equivalente)
- La cola SYSTEM.COMMAND.INPUT y el servidor de mandatos (desde aplicaciones, CSQUTIL, o el conjunto de datos de entrada de inicialización CSQINPX)

Tabla 31. Orígenes desde los que ejecutar mandatos MQSC

Mandato	CSQINP1	CSQINP2	Consola de z/OS	Cola de entrada de mandatos y servidor
ALTER AUTHINFO		X	X	X
ALTER BUFFPOOL		X	X	X
ALTER CFSTRUCT		X	X	X
ALTER CHANNEL		X	X	X
ALTER NAMELIST		X	X	X
ALTER PSID			X	X
ALTER PROCESS		X	X	X
ALTER QALIAS		X	X	X
ALTER QLOCAL		X	X	X
ALTER QMGR		X	X	X
ALTER QMODEL		X	X	X
ALTER QREMOTE		X	X	X

Tabla 31. Orígenes desde los que ejecutar mandatos MQSC (continuación)

Mandato	CSQINP1	CSQINP2	Consola de z/OS	Cola de entrada de mandatos y servidor
ALTER SECURITY	X	X	X	X
ALTER STGCLASS		X	X	X
ALTER SUB		X	X	X
ALTER TOPIC		X	X	X
ALTER TRACE	X	X	X	X
ARCHIVE LOG	X	X	X	X
BACKUP CFSTRUCT			X	X
CLEAR QLOCAL		X	X	X
DEFINE AUTHINFO		X	X	X
DEFINE BUFFPOOL	X	X		
DEFINE CFSTRUCT		X	X	X
DEFINE CHANNEL		X	X	X
DEFINE LOG			X	X
DEFINE NAMELIST		X	X	X
DEFINE PROCESS		X	X	X
DEFINE PSID	X		X	X
DEFINE QALIAS		X	X	X
DEFINE QLOCAL		X	X	X
DEFINE QMODEL		X	X	X
DEFINE QREMOTE		X	X	X
DEFINE STGCLASS		X	X	X
DEFINE SUB			X	X
DEFINE TOPIC		X	X	X
DELETE AUTHINFO		X	X	X
DELETE BUFFPOOL			X	X
DELETE CFSTRUCT		X	X	X
DELETE CHANNEL			X	X
DELETE NAMELIST		X	X	X
DELETE PROCESS		X	X	X
DELETE PSID			X	X
DELETE QALIAS		X	X	X
DELETE QLOCAL		X	X	X

Tabla 31. Orígenes desde los que ejecutar mandatos MQSC (continuación)

Mandato	CSQINP1	CSQINP2	Consola de z/OS	Cola de entrada de mandatos y servidor
DELETE QMODEL		X	X	X
DELETE QREMOTE		X	X	X
DELETE STGCLASS		X	X	X
DELETE SUB		X	X	X
DELETE TOPIC		X	X	X
DISPLAY ARCHIVE	X	X	X	X
DISPLAY AUTHINFO		X	X	X
DISPLAY CFSTATUS			X	X
DISPLAY CFSTRUCT		X	X	X
DISPLAY CHANNEL		X	X	X
DISPLAY CHSTATUS			X	X
DISPLAY CLUSQMGR			X	X
DISPLAY CMDSERV	X	X	X	X
DISPLAY CONN		X	X	X
DISPLAY CHINIT		X	X	X
DISPLAY GROUP		X	X	X
DISPLAY LOG	X	X	X	X
DISPLAY NAMELIST		X	X	X
DISPLAY PROCESS		X	X	X
DISPLAY QALIAS		X	X	X
DISPLAY QCLUSTER		X	X	X
DISPLAY QLOCAL		X	X	X
DISPLAY QMGR		X	X	X
DISPLAY QMODEL		X	X	X
DISPLAY QREMOTE		X	X	X
DISPLAY QSTATUS		X	X	X
DISPLAY QUEUE		X	X	X
DISPLAY SECURITY			X	X
DISPLAY STGCLASS		X	X	X
DISPLAY SUB		X	X	X
DISPLAY TOPIC		X	X	X
DISPLAY SYSTEM	X	X	X	X



Tabla 31. Orígenes desde los que ejecutar mandatos MQSC (continuación)

Mandato	CSQINP1	CSQINP2	Consola de z/OS	Cola de entrada de mandatos y servidor
DISPLAY THREAD		X	X	X
DISPLAY TRACE	X	X	X	X
DISPLAY USAGE		X	X	X
MOVE QLOCAL		X	X	X
PING CHANNEL			X	X
RECOVER BSDS	X	X	X	X
RECOVER CFSTRUCT			X	X
REFRESH CLUSTER		X	X	X
REFRESH QMGR		X	X	X
REFRESH SECURITY		X	X	X
RESET CHANNEL			X	X
RESET CLUSTER		X	X	X
RESET QSTATS		X	X	X
RESET TPIPE			X	X
RESOLVE CHANNEL			X	X
RESOLVE INDOUBT		X	X	X
RESUME QMGR			X	X
RVERIFY SECURITY		X	X	X
SET ARCHIVE	X	X	X	X
SET LOG	X	X	X	X
SET SYSTEM	X	X	X	X
START CHANNEL			X	X
START CHINIT		X	X	X
START CMDSERV	X	X	X	
START LISTENER			X	X
START QMGR			X	
START TRACE	X	X	X	X
STOP CHANNEL			X	X
STOP CHINIT			X	X
STOP CMDSERV	X	X	X	
STOP LISTENER			X	X
STOP QMGR			X	X

Tabla 31. Orígenes desde los que ejecutar mandatos MQSC (continuación)

Mandato	CSQINP1	CSQINP2	Consola de z/OS	Cola de entrada de mandatos y servidor
STOP TRACE	X	X	X	X
SUSPEND QMGR			X	X

En la sección [Mandatos MQSC](#), cada descripción de mandato identifica los orígenes desde los que se puede ejecutar el mandato.

## Mandatos de inicialización de IBM MQ for z/OS

Los mandatos de inicialización se pueden utilizar para controlar el inicio del gestor de colas.

Los mandatos en los conjuntos de datos de entrada de inicialización se procesan cuando IBM MQ se inicializa durante el inicio del gestor de colas. Se pueden emitir tres tipos de mandato desde los conjuntos de datos de entrada de inicialización:

- Mandatos para definir entidades de IBM MQ que no se pueden definir en otro lugar, por ejemplo DEFINE BUFFPOOL.

Estos mandatos deben residir en el conjunto de datos identificado por el nombre DD de CSQINP1. Se procesan antes de la fase de reinicio de la inicialización. No se pueden emitir a través de la consola, paneles de operaciones y de control, o un programa de aplicación. Las respuestas a estos mandatos se graban en el conjunto de datos secuenciales al que hace referencia en la sentencia CSQOUT1 del procedimiento de tarea iniciada.

- Mandatos para definir objetos de IBM MQ que se pueden recuperar tras el reinicio. Estas definiciones se deben especificar en el conjunto de datos identificado por el nombre DD de CSQINP2. Se almacenan en el conjunto de páginas cero. CSQINP2 se procesa después de la fase de reinicio de la inicialización. Las respuestas a estos mandatos se graban en el conjunto de datos secuenciales al que hace referencia en la sentencia CSQOUT2 del procedimiento de tarea iniciada.
- Mandatos para manipular objetos de IBM MQ. Estos mandatos también se deben especificar en el conjunto de datos identificado por el nombre DD de CSQINP2. Por ejemplo, el ejemplo suministrado por IBM MQ contiene un mandato ALTER QMGR para especificar una cola de mensajes no entregados para el subsistema. La respuesta a estos mandatos se graba en el conjunto de datos de salida CSQOUT2.

**Nota:** Si se definen objetos de IBM MQ en CSQINP2, IBM MQ intenta volver a definirlos cada vez que se inicia el gestor de colas. Si los objetos ya existen, el intento de definirlos falla. Si necesita definir los objetos en CSQINP2, puede evitar este problema utilizando el parámetro REPLACE de los mandatos DEFINE, sin embargo, esto altera temporalmente todos los cambios que se realizaron durante la ejecución anterior del gestor de colas.

Con IBM MQ for z/OS se proporcionan miembros de conjunto de datos de inicialización de ejemplo. Se describen en [Definiciones de ejemplo suministradas con IBM MQ](#).

## Mandatos de inicialización para la gestión de colas distribuidas

También puede utilizar el conjunto de datos de inicialización CSQINP2 para el mandato START CHINIT. Si necesita una serie de otros mandatos para definir su entorno de gestión de colas distribuidas (por ejemplo, iniciar los escuchas), IBM MQ proporciona un tercer conjunto de datos de entrada de inicialización, llamado CSQINPX, que se procesa como parte del procedimiento de tarea iniciada del iniciador de canal.

Los mandatos MQSC contenidos en el conjunto de datos se ejecutan al final de la inicialización del iniciador de canal, y la salida se graba en el conjunto de datos especificado mediante la sentencia

CSQOUTX DD. Puede utilizar el conjunto de datos de inicialización CSQINPX para iniciar los escuchas, por ejemplo.

Con IBM MQ for z/OS se proporciona un miembro de conjunto de datos de inicialización de iniciador de canal de ejemplo. Se describe en [Definiciones de ejemplo suministradas con IBM MQ](#).

## Mandatos de inicialización para publicación/suscripción

Si necesita una serie de mandatos para definir el entorno de publicación/suscripción (por ejemplo, al definir suscripciones), IBM MQ proporciona un cuarto conjunto de datos de entrada de inicialización, denominado CSQINPT.

Los mandatos MQSC contenidos en el conjunto de datos se ejecutan al final de la inicialización de publicación/suscripción, y la salida se escribe en el conjunto de datos especificado mediante la sentencia CSQOUTT DD. Puede utilizar el conjunto de datos de inicialización CSQINPT para definir suscripciones, por ejemplo.

Con IBM MQ for z/OS se proporciona un miembro de conjunto de datos de inicialización de publicación/suscripción de ejemplo. Se describe en [Definiciones de ejemplo suministradas con IBM MQ](#).

z/OS

## Los programas de utilidad de IBM MQ for z/OS

IBM MQ for z/OS proporciona un conjunto de programas de utilidad que puede utilizar para ayudar en la administración del sistema.

IBM MQ for z/OS proporciona un conjunto de programas de utilidad para ayudarle a realizar diversas tareas administrativas, incluidas las siguientes:

- Gestionar políticas de seguridad de mensajes.
- Realizar copia de seguridad, restauración, y reorganización de las tareas.
- Emitir mandatos y definiciones de objetos de proceso.
- Generar salida de conversión de datos.
- Modificar el conjunto de datos de rutina de carga.
- Listar información sobre los registros.
- Imprimir los registros.
- Configurar tablas de Db2 y otros programas de utilidad de Db2.
- Procesar mensajes de la cola de mensajes no entregados.

### El programa de utilidad de políticas de seguridad de mensajes

El programa de utilidad de políticas de seguridad de mensajes (CSQOUTIL) se ejecuta como un programa de utilidad autónomo para gestionar políticas de seguridad de mensajes. Consulte [El programa de utilidad de políticas de seguridad de mensajes \(CSQOUTIL\)](#) para obtener más información.

### El programa de utilidad CSQUTIL

Este es un programa de utilidad proporcionado para ayudarle con las tareas de copia de seguridad, restauración y reorganización. Consulte [El programa de utilidad CSQUTIL](#) si desea más información.

### El programa de utilidad de conversión de datos

El programa de utilidad de salida de conversión de datos de IBM MQ for z/OS (**CSQUCVX**) se ejecuta como un programa de utilidad autónomo para crear rutinas de salida de conversión de datos.

### El programa de utilidad de inventario de registro de cambios

El programa de utilidad de inventario de registro de cambios de IBM MQ for z/OS (**CSQJU003**) se ejecuta como un programa de utilidad autónomo para cambiar el conjunto de datos de rutina de carga (BSDS). Puede utilizar el programa de utilidad para realizar las funciones siguientes:

- Añadir o suprimir conjuntos de datos del registro de archivado o activo.
- Suministrar contraseñas para registros de archivado.

## El programa de utilidad de correlación de registro de impresión

El programa de utilidad de correlación de registro de impresión de IBM MQ for z/OS (**CSQJU004**) se ejecuta como un programa de utilidad autónomo para listar la siguiente información:

- Nombre del conjunto de datos de registro y asociación RBA de registro para las copias de todos los conjuntos de datos de registro de archivado y activos. Si el registro dual no está activo, sólo hay una copia de los conjuntos de datos.
- Los conjuntos de datos de registro activos disponibles para nuevos datos de registro.
- Contenido de la cola de registros de puntos de comprobación en el conjunto de datos de la rutina de carga (BSDS).
- Contenido del registro de historial de mandatos del registro de archivado.
- Indicaciones de fecha y hora del sistema y del programa de utilidad.

## El programa de utilidad de impresión de registro

Programa de utilidad de impresión de registro (**CSQ1LOGP**) se ejecuta como un programa de utilidad autónomo. Puede ejecutar el programa de utilidad especificando:

- Un conjunto de datos de rutina de carga (BSDS)
- Registros activos (sin ningún BSDS)
- Registros de archivado (sin ningún BSDS)

## Programa de utilidad de grupo de compartición de colas

El programa de utilidad de grupo de compartición de colas (**CSQ5PQSG**) se ejecuta como un programa de utilidad autónomo para configurar las tablas de Db2 y realizar otras tareas de Db2 necesarias para los grupos de compartición de colas.

## El programa de utilidad de preformato de registro activo

El programa de utilidad de preformato de registro activo (**CSQJUFMT**) formatea conjuntos de datos de registro activo antes de que el gestor de colas los utilice. Si el programa de utilidad preformatea los conjuntos de datos de registro activo, se mejora el rendimiento de grabación de registro en la primera pasada por los registros activos del gestor de colas.

## El programa de utilidad de manejador de la cola de mensajes no entregados

El programa de utilidad de manejador de la cola de mensajes no entregados (**CSQUDLQH**) se ejecuta como un programa de utilidad autónomo. Comprueba los mensajes que están en la cola de mensajes no entregados y los procesa de acuerdo con un conjunto de reglas que se suministran al programa de utilidad.

## El programa de utilidad qload

A partir de IBM MQ 8.0, el programa de utilidad **qload**, que se entrega en IBM MQ Supportpac MO03, se ha integrado en IBM MQ como el programa de utilidad de **dmpmqmsg**.

En z/OS, el programa de utilidad está disponible como un módulo ejecutable, CSQUDMSG en la biblioteca SCSQLOAD, con un alias de QLOAD para la compatibilidad. También se proporciona un JCL de ejemplo como miembro CSQ4QLOD de SCSQPROC.

## Utilidad CSQUTIL de IBM MQ for z/OS

El programa de utilidad CSQUTIL se proporciona con IBM MQ for z/OS para ayudarle a realizar tareas de copia de seguridad, restauración y reorganización, y para emitir mandatos y procesar definiciones de objetos.

Para obtener más información sobre el programa de utilidad CSQUTIL, consulte [Programa de utilidad de IBM MQ \(CSQUTIL\)](#). Mediante este programa de utilidad, puede invocar las funciones siguientes:

### **command**

Para emitir mandatos MQSC, grabar las definiciones de objeto y crear archivos de definiciones de canal de cliente.

### **COPY**

Para leer el contenido de una cola de mensajes de IBM MQ for z/OS especificada o el contenido de todas las colas de un conjunto de páginas especificado y colocarlo en un archivo secuencial y conservar la cola original.

### **COPYPAGE**

Para copiar conjuntos de páginas completos en conjuntos de páginas más grandes.

### **EMPTY**

Para suprimir el contenido de una cola de mensajes de IBM MQ for z/OS especificada o el contenido de todas las colas de un conjunto de páginas especificado, reteniendo las definiciones de las colas.

### **Formato**

Para dar formato a conjuntos de páginas de IBM MQ for z/OS.

### **LOAD**

Para restaurar el contenido de una cola de mensajes de IBM MQ for z/OS especificada o el contenido de todas las colas de un conjunto de páginas especificado de un archivo secuencial creado por la función COPY.

### **PAGEINFO**

Para extraer información de conjunto de páginas de uno o más conjuntos de páginas.

### **RESETPAGE**

Para copiar conjuntos de páginas enteros en otros conjuntos de datos de conjuntos de páginas y restablecer la información de registro en la copia.

### **SCOPY**

Para copiar el contenido de una cola en un conjunto de datos mientras el gestor de colas está fuera de línea.

### **SDEFS**

Para generar un conjunto de mandatos DEFINE para objetos mientras el gestor de colas está fuera de línea.

### **SLOAD**

Para restaurar mensajes del conjunto de datos de destino de una operación COPY o SCOPY anterior. SLOAD procesa una sola cola.

### **SWITCH**

Para cambiar o consultar la cola de transmisión asociada con los canales de clúster emisor.

### **XPARM**

Para convertir un módulo de carga de parámetros del iniciador de canal en atributos del gestor de colas (para fines de migración).

## Operando IBM MQ for z/OS

Siga estos procedimientos básicos para utilizar IBM MQ for z/OS.

Puede realizar las operaciones descritas en esta sección utilizando el IBM MQ Explorer, que se distribuye con IBM MQ for Windows y IBM MQ para Linux (plataformas x86 y x86-64). Está disponible un IBM MQ Explorer autónomo para descargar desde Fix Central. Para obtener más información, consulte [“Administración utilizando IBM MQ Explorer”](#) en la página 133.

## Ejecución de mandatos de gestor de colas en z/OS

Puede emitir mandatos de control de IBM MQ desde una consola z/OS o con el programa de utilidad CSQUTIL. Los mandatos pueden utilizar una serie de prefijo de mandato (CPF) para indicar qué subsistema de IBM MQ procesa el mandato.

Puede controlar la mayor parte del entorno operativo de IBM MQ mediante los mandatos de IBM MQ. IBM MQ for z/OS da soporte a los tipos MQSC y PCF de estos mandatos. En este tema se describe cómo especificar atributos utilizando mandatos MQSC, y por ello hace referencia a los mandatos y atributos utilizando sus nombres de mandato MQSC, en lugar de sus nombres PCF. Para obtener información detallada de la sintaxis de los mandatos MQSC, consulte [Mandatos MQSC](#). Para obtener detalles de la sintaxis de los mandatos PCF, consulte [“Utilización de los formatos de mandato programable de IBM MQ”](#) en la página 26. Si es un usuario debidamente autorizado, puede emitir mandatos de IBM MQ desde:

- Los conjuntos de datos de entrada de inicialización (descritos en [“Mandatos de inicialización de IBM MQ for z/OS”](#) en la página 442).
- Una consola de z/OS o equivalente, como SDSF
- La rutina de mandato get maestra de z/OS, MGCRE (SVC 34)
- El programa de utilidad CSQUTIL de IBM MQ (descrito en [Programa de utilidad de IBM MQ](#)).
- Una aplicación de usuario, que puede ser:
  - Un programa CICS
  - Un programa TSO
  - Un programa por lotes z/OS
  - Un programa IMS

Consulte [“Escribir programas para administrar IBM MQ for z/OS”](#) en la página 467 para obtener más información al respecto.

Gran parte de la funcionalidad de estos mandatos se proporciona de forma conveniente mediante los paneles de operaciones y paneles de control, accesibles desde TSO e ISPF, y se describe en [“Operaciones y paneles de control de IBM MQ for z/OS”](#) en la página 452.

Para obtener más información, consulte

- [“Emisión de mandatos desde una consola z/OS o su equivalente”](#) en la página 446
  - [Series de prefijo de mandato](#)
  - [Utilización de la consola de z/OS para emitir mandatos](#)
  - [Respuestas de mandatos](#)
- [Emisión de mandatos desde el programa de utilidad CSQUTIL](#)

### Emisión de mandatos desde una consola z/OS o su equivalente

Puede emitir todos los mandatos de IBM MQ desde una consola z/OS o su equivalente. También puede emitir mandatos de IBM MQ desde cualquier lugar donde pueda emitir mandatos de z/OS, como SDSF o con un programa que utilice la macro MGCRE.

La cantidad máxima de datos que se pueden visualizar como resultado de un mandato escrito en la consola es 32 KB.

**Nota:**

1. No puede emitir mandatos de IBM MQ que utilicen el formato de mandato IMS/SSR desde un terminal IMS. Esta función no está soportada por el adaptador IMS.
2. El campo de entrada proporcionado por SDSF podría no ser lo suficientemente largo para algunos mandatos, especialmente los mandatos para canales.

**Series de prefijo de mandato**

Cada mandato de IBM MQ debe ir precedido de una serie de prefijo de mandato (CPF), tal como se muestra en la [Figura 35 en la página 447](#).

Puesto que puede haber más de un subsistema de IBM MQ ejecutándose en z/OS, la CPF se utiliza para indicar qué subsistema de IBM MQ procesa el mandato. Por ejemplo, para iniciar el gestor de colas para un subsistema llamado CSQ1, donde la CPF es '+CSQ1', emita el mandato '+CSQ1 START QMGR' desde la consola del operador. Esta CPF debe estar definida en la tabla de nombres de subsistema (para el subsistema CSQ1). Esto se describe en [Definir series de prefijo de mandatos \(CPF\)](#). En los ejemplos, la serie '+CSQ1' se utiliza como el prefijo de mandato.

**Utilización de la consola de z/OS para emitir mandatos**

Puede escribir mandatos simples desde la consola de z/OS, por ejemplo, el mandato DISPLAY en la [Figura 35 en la página 447](#). Sin embargo, para mandatos complejos o para conjuntos de mandatos que emita con frecuencia, los otros métodos de emisión de mandatos son mejores.

```
+CSQ1 DISPLAY QUEUE(TRANSMIT.QUEUE.PROD) TYPE(QLLOCAL)
```

*Figura 35. Emitir un mandato DISPLAY desde la consola de z/OS*

**Respuestas de mandatos**

Se envían respuestas directas a mandatos a la consola que ha emitido el mandato. IBM MQ da soporte a la función *Soporte de consola ampliado* (EMCS) disponible en z/OSy, por lo tanto, se pueden utilizar consolas con ID de 4 bytes. Además, todos los mandatos, excepto START QMGR y STOP QMGR, dan soporte al uso de Señales de mandato y respuesta (CARTs) cuando el mandato se emite con un programa que utiliza la macro MGCRC.

**Emisión de mandatos desde el programa de utilidad CSQUTIL**

Puede emitir mandatos desde un conjunto de datos secuenciales utilizando la función COMMAND del programa de utilidad CSQUTIL. Este programa de utilidad transfiere los mandatos, como mensajes, a la *cola de entrada de mandatos del sistema* y espera la respuesta, que se imprime junto con los mandatos originales en SYSPRINT. Para obtener información detallada al respecto, consulte [Programa de utilidad de IBM MQ](#).

**z/OS Inicio y parada de un gestor de colas en z/OS**

Utilice este tema como introducción para detener e iniciar un gestor de colas.

Esta sección describe cómo iniciar y detener un gestor de colas. Contiene información sobre los temas siguientes:

- [“Antes de iniciar IBM MQ” en la página 448](#)
- [“Inicio de un gestor de colas” en la página 448](#)
- [“Detención de un gestor de colas” en la página 450](#)

Iniciar y detener un gestor de colas es relativamente sencillo. Cuando un gestor de colas se detiene bajo condiciones normales, su última acción es tomar un punto de comprobación de terminación. Este punto de comprobación y los registros proporcionan al gestor de colas la información que necesita para reiniciar.

Esta sección contiene información sobre los mandatos START y STOP y contiene una breve visión general del inicio después de que se haya producido una terminación anómala.

## Antes de iniciar IBM MQ

Después de haber instalado IBM MQ, éste se define como un subsistema z/OS formal. Este mensaje aparece durante cualquier carga del programa inicial (IPL) de z/OS:

```
CSQ3110I +CSQ1 CSQ3UR00 - SUBSYSTEM ssnm INITIALIZATION COMPLETE
```

donde *ssnm* es el nombre de subsistema IBM MQ.

De ahora en adelante, puede iniciar el gestor de colas para dicho subsistema *desde cualquier consola de z/OS que haya sido autorizada para emitir mandatos de control del sistema*; es decir, un grupo de mandatos SYS de z/OS. Debe emitir el mandato START desde la consola autorizada, no puede emitirlo a través de JES o TSO.

Si utiliza grupos de compartición de colas, debe iniciar primero RRS y luego Db2, antes de iniciar el gestor de colas.

## Inicio de un gestor de colas

Inicie un gestor de colas emitiendo un mandato START QMGR. Sin embargo, no puede utilizar correctamente el mandato START a menos que tenga la autorización apropiada. Consulte [Configuración de la seguridad en z/OS](#) para obtener información sobre la seguridad de IBM MQ. La [Figura 36 en la página 448](#) muestra ejemplos del mandato START. (Recuerde que debe anteponer una serie de prefijo de mandato (CPF) a un mandato de IBM MQ.)

```
+CSQ1 START QMGR
+CSQ1 START QMGR PARM(NEWLOG)
```

*Figura 36. Inicio del gestor de colas desde una consola de z/OS*

Consulte [START QMGR](#) para obtener información sobre la sintaxis del mandato START QMGR.

No puede ejecutar el gestor de colas como un trabajo por lotes o iniciarlo mediante el mandato de z/OS START. Estos métodos probablemente iniciarán un espacio de direcciones para IBM MQ que luego finaliza de forma anómala. Tampoco puede iniciar un gestor de colas desde el programa de utilidad CSQUTIL o una aplicación de usuario similar.

Sin embargo, puede iniciar un gestor de colas desde un programa autorizado para APF pasando un mandato START QMGR al servicio de z/OS MGCRE (SVC 34).

Si utiliza grupos de compartición de colas, los sistemas Db2 asociados y RRS deben estar activos cuando inicie el gestor de colas.



## Opciones de inicio

Cuando inicie un gestor de colas, se carga un módulo de parámetro de sistema. Puede especificar el nombre del módulo de parámetro del sistema de una de estas dos formas:

- Con el parámetro PARM del mandato /cpf START QMGR, por ejemplo

```
/cpf START QMGR PARM(CSQ1ZPRM)
```

- Con un parámetro en el procedimiento de inicio, por ejemplo, codifique la sentencia JCL EXEC como

```
//MQM EXEC PGM=CSQYASCP,PARM='ZPARAM(CSQ1ZPRM)'
```

Un módulo de parámetro del sistema proporciona información especificada cuando se personalizó el gestor de colas.

**V 9.1.0** A partir de IBM MQ 9.1.0, puede utilizar la opción **QMGRPROD** para especificar el producto con el que se registrará el uso del gestor de colas y la opción **AMSPROD** para especificar el equivalente para AMS si se utiliza. Consulte el mandato [START QMGR](#) de MQSC para obtener detalles sobre los valores permitidos.

**V 9.1.0** A continuación, se muestra un ejemplo de una sentencia JCL EXEC:

```
//MQM EXEC PGM=CSQYASCP,PARM='QMGRPROD(MQ)'
```

Consulte [z/OS Gestión de productos MVS](#) para obtener más información sobre la grabación de uso del producto.

También puede utilizar la opción ENVPARM para sustituir uno o más parámetros en el procedimiento de JCL para el gestor de colas.

Por ejemplo, puede actualizar el procedimiento de inicio del gestor de colas, de modo que el DDname CSQINP2 es una variable. Esto significa que puede cambiar el CSQINP2 DDname sin cambiar el procedimiento de inicio. Esto es útil para implementar los cambios, proporcionando restituciones para operadores y operaciones de gestor de colas.

Suponga que el procedimiento de inicio para el gestor de colas CSQ1 tenía un aspecto como en de la [Figura 37](#) en la página 449.

```
//CSQ1MSTR PROC INP2=NORM
//MQMESA EXEC PGM=CSQYASCP
//STEPLIB DD DISP=SHR,DSN=thlqual.SCSQANLE
// DD DISP=SHR,DSN=thlqual.SCSQAUTH
// DD DISP=SHR,DSN=db2qual.SDSNLOAD
//BSDS1 DD DISP=SHR,DSN=myqual.BSDS01
//BSDS2 DD DISP=SHR,DSN=myqual.BSDS02
//CSQP0000 DD DISP=SHR,DSN=myqual.PSID00
//CSQP0001 DD DISP=SHR,DSN=myqual.PSID01
//CSQP0002 DD DISP=SHR,DSN=myqual.PSID02
//CSQP0003 DD DISP=SHR,DSN=myqual.PSID03
//CSQINP1 DD DISP=SHR,DSN=myqual.CSQINP(CSQ1INP1)
//CSQINP2 DD DISP=SHR,DSN=myqual.CSQINP(CSQ1&INP2.)
//CSQOUT1 DD SYSOUT=*
//CSQOUT2 DD SYSOUT=*
```

*Figura 37. Procedimiento de inicio de ejemplo*

Si, a continuación, inicia el gestor de colas con el mandato:

```
+CSQ1 START QMGR
```

el CSQINP2 utilizado es un miembro llamado CSQ1NORM.

Sin embargo, suponga que está colocando un nuevo conjunto de programas en producción, de forma que la próxima vez que inicie el gestor de colas CSQ1, las definiciones de CSQINP2 se tomarán del miembro CSQ1NEW. Para hacer esto, tendría que iniciar el gestor de colas con este mandato:

```
+CSQ1  START QMGR ENVPARM('INP2=NEW')
```

y CSQ1NEW se utilizaría en lugar de CSQ1NORM. Nota: z/OS limita las especificaciones de KEYWORD=value para los parámetros simbólicos (como en INP2=NEW) a 255 caracteres.

### Inicio después de una terminación anómala

IBM MQ detecta automáticamente si el reinicio sigue a una conclusión normal o a una terminación anómala.

El inicio de un gestor de colas después de que termine de forma anómala es diferente del inicio después de que se haya emitido el mandato STOP QMGR. Tras STOP QMGR, el sistema finaliza su trabajo de una forma ordenada y toma un punto de comprobación de terminación después de detenerse. Cuando se reinicia el gestor de colas, éste utiliza información del registro de recuperación y de punto de comprobación del sistema para determinar el estado del sistema en la conclusión.

Sin embargo, si el gestor de colas termina de forma anómala, termina sin poder finalizar su trabajo o tomando un punto de comprobación de terminación. Cuando reinicie un gestor de colas después de una terminación anómala, renueva el conocimiento de su estado durante la terminación utilizando la información del registro y le notifica el estado de distintas tareas. Normalmente, el proceso de reinicio resuelve todos los estados incoherentes. No obstante, en algunos casos, deberá tomar medidas concretas para resolver las incoherencias.

### Mensajes de usuario durante el inicio

Cuando inicia un gestor de colas correctamente, el gestor de colas genera un conjunto de mensajes de inicio.

### Detención de un gestor de colas

Antes de detener un gestor de colas, todos los mensajes WTOR (WTOR) (escribir-a-operador-con-respuesta) relacionados con IBM MQ deben recibir respuestas, por ejemplo, obtener solicitudes de registro. Cada mandato de la [Figura 38 en la página 450](#) termina un gestor de colas en ejecución.

```
+CSQ1  STOP QMGR
+CSQ1  STOP QMGR MODE(QUIESCE)
+CSQ1  STOP QMGR MODE(FORCE)
+CSQ1  STOP QMGR MODE(RESTART)
```

*Figura 38. Detención de un gestor de colas*

El mandato STOP QMGR toma como valor predeterminado STOP QMGR MODE(QUIESCE).

En la modalidad QUIESCE (de desactivación temporal), IBM MQ no permite que se creen nuevas hebras de conexión, pero permite a las hebras existentes continuar; sólo termina cuando todas las hebras han finalizado. Las aplicaciones pueden solicitar que se les notifique en el caso de la desactivación temporal del gestor de colas. Por lo tanto, utilice la modalidad QUIESCE cuando sea posible para que las

aplicaciones que han solicitado una notificación tengan la oportunidad de desconectarse. Consulte [Qué ocurre durante la terminación](#) para obtener detalles.

Si el gestor de colas no termina en un tiempo razonable en respuesta a un mandato STOP QMGR MODE(QUIESCE), utilice el mandato DISPLAY CONN para determinar si existe alguna hebra de conexión y emprenda los pasos necesarios para terminar las aplicaciones asociadas. Si no hay ninguna hebra, emita un mandato STOP QMGR MODE(FORCE).

Los mandatos STOP QMGR MODE(QUIESCE) y STOP QMGR MODE(FORCE) anulan el registro de IBM MQ del Gestor de reinicio automático (ARM) de MVS, lo que impide que el ARM reinicie el gestor de colas automáticamente. El mandato STOP QMGR MODE(RESTART) funciona del mismo modo que el mandato STOP QMGR MODE(FORCE), salvo que no anula el registro de IBM MQ del ARM. Esto significa que el gestor de colas es elegible para el reinicio automático inmediato.

Si el subsistema de IBM MQ no está registrado en el ARM, el mandato STOP QMGR MODE(RESTART) es rechazado y el siguiente mensaje se envía a la consola de z/OS:

```
CSQY205I ARM element arm-element is not registered
```

Si este mensaje no se emite, el gestor de colas se reinicia automáticamente. Para obtener más información sobre ARM, consulte [“Utilización del Gestor de reinicio automático \(ARM\) de z/OS”](#) en la página 530.

### **Cancele el espacio de direcciones del gestor de colas sólo si STOP QMGR MODE(FORCE) no termina el gestor de colas.**

Si un gestor de colas es detenido por la cancelación del espacio de direcciones o por el uso del mandato STOP QMGR MODE(FORCE), se mantiene la coherencia con los sistemas CICS o IMS conectados. La resincronización de los recursos se inicia cuando se reinicia un gestor de colas y se completa cuando se establece la conexión con el sistema CICS o IMS.

**Nota:** Cuando detenga el gestor de colas, es posible que descubra que se ha emitido el mensaje IEF352I. z/OS emite este mensaje si detecta que el no marcar el espacio de direcciones como no utilizable daría lugar a un riesgo para la integridad. Puede ignorar este mensaje.

### **Detener mensajes**

Tras emitir un mandato STOP QMGR, podrá obtener los mensajes CSQY009I y CSQY002I, por ejemplo:

```
CSQY009I +CSQ1 ' STOP QMGR' COMMAND ACCEPTED FROM  
USER(userid), STOP MODE(FORCE)  
CSQY002I +CSQ1 QUEUE MANAGER STOPPING
```

Donde `userid` es el ID de usuario que ha emitido el mandato STOP QMGR y el parámetro MODE depende del especificado en el mandato.

Cuando el mandato STOP se ha completado correctamente, se visualizan los mensajes siguientes en la consola de z/OS:

```
CSQ9022I +CSQ1 CSQYASCP ' STOP QMGR' NORMAL COMPLETION  
CSQ3104I +CSQ1 CSQ3EC0X - TERMINATION COMPLETE
```

Si utiliza el ARM y no ha especificado MODE(RESTART), también se visualiza el siguiente mensaje:

```
CSQY204I +CSQ1 ARM DEREGISTER for element arm-element type  
arm-element-type successful
```

No puede reiniciar el gestor de colas hasta que se haya visualizado el siguiente mensaje:

```
CSQ3100I +CSQ1 CSQ3EC0X - SUBSYSTEM ssnm READY FOR START COMMAND
```

## Operaciones y paneles de control de IBM MQ for z/OS

Puede utilizar los paneles de operaciones y los paneles de control de IBM MQ para realizar tareas de administración en objetos de IBM MQ. Utilice este tema como introducción a los mandatos y los paneles de control.

Estos paneles se utilizan para definir, visualizar, modificar o suprimir objetos de IBM MQ. Utilice los paneles para la administración diaria y para realizar pequeños cambios en los objetos. Si va a configurar o cambiar muchos objetos, utilice la función COMMAND del programa de utilidad CSQUTIL.

Los paneles de operaciones y los paneles de control dan soporte a los controles para el iniciador de canal (por ejemplo, para iniciar un canal o un escucha TCP/IP), para la agrupación en clúster y para la seguridad. También le permiten visualizar información sobre el uso de conjuntos de páginas y hebras.

Los paneles funcionan mediante el envío de mandatos de IBM MQ de tipo MQSC a un gestor de colas, a través de la cola de entrada de mandatos del sistema.

### Nota:

1. Es posible que las operaciones y los paneles de control de z/OS IBM MQ (CSQOREXX) no den soporte a las nuevas funciones y parámetros añadidos a partir de la versión 7. Por ejemplo, no hay paneles para la manipulación directa de suscripciones u objetos de tema.

Si utiliza uno de los siguientes mecanismos soportados podrá administrar definiciones de publicación/suscripción y otros controles del sistema que no están disponibles directamente en otros paneles.

- a. IBM MQ Explorer
- b. Consola de z/OS
- c. Mensajes PCF (Programmable Command Format)
- d. Función COMMAND de CSQUTIL
- e. Consola web de IBM MQ

Tenga en cuenta que la acción **Command** genérica de los paneles CSQOREXX le permite emitir cualquier mandato MQSC válido, incluidos mandatos relacionados con SMDS. Puede utilizar todos los mandatos que emite la función COMMAND de CSQUTIL.

2. No puede emitir los mandatos de IBM MQ directamente desde la línea de mandatos de los paneles.
3. Para utilizar los paneles de operaciones y los paneles de control, debe tener la autorización de seguridad correcta; esto se describe en [Identificadores de usuario para la seguridad de mandatos y de recursos de mandatos](#).
4. No puede proporcionar un ID de usuario y contraseña utilizando CSQUTIL o los paneles CSQOREXX. En su lugar, si el ID de usuario tiene autorización UPDATE para el perfil BATCH en MQCONN, puede ignorar el valor **CHKLOCL**(REQUIRED. Consulte [Utilización de CHKLOCL en aplicaciones enlazadas localmente](#) para obtener más información.

## Invocación y reglas para los paneles de operaciones y los paneles de control

Puede controlar IBM MQ y emitir mandatos de control mediante los paneles ISPF.

## Cómo acceder a los paneles de operaciones y de control de IBM MQ

Si el menú de opciones principal ISPF/PDF se ha actualizado para IBM MQ, puede acceder a los paneles de operaciones y los paneles de control de IBM MQ desde ese menú. Si desea información detallada sobre la actualización del menú, consulte [Tarea 20: Configurar las operaciones y los paneles de control](#).

Puede acceder a los paneles de operaciones y los paneles de control de IBM MQ desde el panel del procesador de mandatos TSO (normalmente la opción 6 en el menú de opciones principal ISPF/PDF). El nombre del exec que se ejecuta para hacer esto es CSQOREXX. Tiene dos parámetros; `thlqual` es el calificador de alto nivel para las bibliotecas de IBM MQ que se van a utilizar, y `langletter` es la carta que identifica las bibliotecas de idioma nacional que se utilizarán (por ejemplo, E para EE.UU. Inglés). Los parámetros se pueden omitir si las bibliotecas de IBM MQ están instaladas permanentemente en la configuración de ISPF. Como alternativa, puede emitir CSQOREXX desde la línea de mandatos de TSO.

Estos paneles están diseñados para ser utilizados por operadores y administradores con un mínimo de formación profesional. Lea estas instrucciones con los paneles en ejecución y pruebe las diferentes tareas sugeridas.

**Nota:** Al utilizar los paneles, las colas dinámicas temporales con nombres con el formato SYSTEM.CSQOREXX.\* se crean.

## Reglas para los paneles de operaciones y los paneles de control

Consulte [Reglas de denominación de IBM MQ objetos](#) sobre las reglas generales para las series de caracteres y nombres de IBM MQ. Sin embargo, hay algunas reglas que sólo se aplican a los paneles de operaciones y los paneles de control:

- No encierre las series, por ejemplo descripciones, entre comillas simples o dobles.
- Si incluye un apóstrofo o comilla en un campo de texto, no es necesario que lo repita ni que añada un carácter de escape. Los caracteres se guardan exactamente como se escriben; por ejemplo:

```
This is Maria's queue
```

El procesador del panel los duplica automáticamente para pasarlos a IBM MQ. Sin embargo, si tiene que truncar los datos para ello, lo hace.

- Puede utilizar caracteres en mayúsculas o minúsculas en la mayoría de los campos, y estos se convierten a caracteres en mayúsculas cuando se pulsa Intro. Existen las siguientes excepciones:
  - Nombres de clase de almacenamiento y nombres de estructura de recurso de acoplamiento, que deben empezar por mayúsculas de la A a la Z e ir seguidos de mayúsculas de la A a la Z o caracteres numéricos.
  - Determinados campos que no se traducen. Incluyen los siguientes:
    - ID de aplicación
    - Descripción
    - Datos de entorno
    - Nombres de objeto (pero si utiliza un nombre de objeto en minúsculas, es posible que no pueda especificarlo en una consola de z/OS)
    - Nombre del sistema remoto
    - Datos desencadenantes
    - Datos de usuario
- En los nombres, los espacios en blanco iniciales y los caracteres de subrayado iniciales se ignoran. Por lo tanto, no puede tener nombres de objeto que empiecen por espacios en blanco o caracteres de subrayado.

- Se utilizan caracteres de subrayado para mostrar la extensión de los campos en blanco. Cuando se pulsa Intro, los caracteres de subrayado finales se sustituyen por espacios en blanco.
- Muchos campos de descripción y de texto se presentan en varias partes, y IBM MQ maneja cada parte de forma independiente. Esto significa que los espacios en blanco finales se mantienen y el texto no es contiguo.

## Campos en blanco

Cuando se especifica la acción **Definir** para un objeto de IBM MQ, cada campo del panel de definición contiene un valor. Consulte la ayuda general (ayuda ampliada) para los paneles de visualización para obtener información sobre la ubicación en la que IBM MQ obtiene los valores. Si escribe encima de un campo con espacios en blanco, y no están permitidos los espacios en blanco, IBM MQ coloca el valor predeterminado de la instalación en el campo o le solicita que entre el valor requerido.

Cuando se especifica la acción **Modificar** para un objeto de IBM MQ, cada campo del panel de modificación contiene el valor actual de ese campo. Si escribe encima de un campo con espacios en blanco, y no están permitidos espacios en blanco, el valor de ese campo no se modifica.

## **Objetos y acciones en z/OS**

Los paneles de operaciones y los paneles de control le ofrecen muchos tipos de objeto diferentes y una serie de acciones que puede realizar en ellos.

Las acciones se enumeran en el panel inicial y le permiten manipular los objetos y visualizar información sobre ellos. Estos objetos incluyen todos los objetos de IBM MQ, así como algunos adicionales. Los objetos se dividen en las siguientes categorías.

- [Colas, procesos, objetos de información de autenticación, listas de nombres, clases de almacenamiento y estructuras CF](#)
- [Canales](#)
- [Objetos de clúster](#)
- [Gestor de colas y seguridad](#)
- [Conexiones](#)
- [system](#)

Consulte Acciones para ver una tabla de referencias cruzadas de las acciones que se pueden realizar en los objetos de IBM MQ.

### **Colas, procesos, objetos de información de autenticación, listas de nombres, clases de almacenamiento y estructuras CF**

Estos son los objetos básicos de IBM MQ. Puede haber muchos de cada tipo. Se pueden listar, listar con filtro, definir y suprimir, y tienen atributos que se pueden visualizar y modificar, utilizando las acciones LIST o DISPLAY, LIST con FILTER, DEFINE LIKE, MANAGE y ALTER. (Los objetos se suprimen utilizando la acción MANAGE.)

Esta categoría consta de los siguientes objetos:

QLOCAL	Cola local
QREMOTE	Cola remota
QALIAS	Cola alias para referencia indirecta a una cola
QMODEL	Cola modelo para definir colas dinámicamente
QUEUE	Cualquier tipo de cola
QSTATUS	Estado de una cola local

PROCESS	Información sobre una aplicación que se debe iniciar cuando se produce un suceso desencadenante
AUTHINFO	Información de autenticación: las definiciones necesarias para llevar a cabo la comprobación de CRL (Lista de revocación de certificados) utilizando servidores LDAP
LISTA DE NOMBRES	Lista de nombres, como colas o clústeres
STGCLASS	Clase de almacenamiento
CFSTRUCT	estructura de recurso de acoplamiento (CF)
CFSTATUS	Estado de una estructura CF

## Canales

Los canales se utilizan para la gestión de colas distribuidas. Puede haber muchos de cada tipo, y se pueden listar, listar con filtro, definir, suprimir, visualizar y modificar. También tienen otras funciones disponibles utilizando las acciones START, STOP y PERFORM. PERFORM proporciona las funciones restablecer, ping y resolver canal.

Esta categoría consta de los siguientes objetos:

CHANNEL	Cualquier tipo de canal
SENDER	Canal emisor
Servidor	Canal servidor
RECEIVER	Canal receptor
REQUESTER	Canal peticionario
CLUSRCVR	Canal de clúster receptor
CLUSDR	Canal de clúster emisor
SVRCONN	Canal de conexión con el servidor
CLNTCONN	Canal de conexión con el cliente
CHSTATUS	Estado de una conexión de canal

## Objetos de clúster

Los objetos de clúster se crean automáticamente para las colas y canales que pertenecen a un clúster. Las definiciones de cola base y de canal pueden estar en otro gestor de colas. Puede haber muchos de cada tipo, y los nombres pueden estar duplicados. Estos objetos se pueden listar, listar con filtro y visualizar. PERFORM, START y STOP también están disponibles mediante las acciones LIST.

Esta categoría consta de los siguientes objetos:

CLUSQ	Cola de clúster, creada para una cola que pertenece a un clúster
CLUSCHL	Canal de clúster, creado para un canal que pertenece a un clúster
CLUSQMGR	Gestor de colas de clúster, lo mismo que un canal de clúster pero identificado por su nombre de gestor de colas

Los canales de clúster y los gestores de colas de clúster tienen las acciones PERFORM, START y STOP, pero sólo indirectamente a través de la acción DISPLAY.

## Gestor de colas y seguridad

Los objetos de gestor de colas y de seguridad tienen una sola instancia. Se pueden listar, tienen atributos que se pueden visualizar y modificar (utilizando las acciones LIST o DISPLAY y ALTER), y tienen otras funciones disponibles mediante la acción PERFORM.

Esta categoría consta de los siguientes objetos:

MANAGER	Gestor de colas: la acción PERFORM proporciona las funciones suspender y reanudar clúster
seguridad	Funciones de seguridad: la acción PERFORM proporciona las funciones renovar y volver a verificar

### Conexión

Las conexiones se pueden listar, listar con filtro y visualizar.

Esta categoría consta únicamente del objeto de conexión, CONNECT

### Sistema

Una colección de otras funciones. Esta categoría consta de los siguientes objetos:

SYSTEM	Funciones de sistema
CONTROL	Sinónimo de SYSTEM

Las funciones disponibles son:

LIST o DISPLAY	Visualizar información de uso de grupo de compartición de colas, colas distribuidas, conjunto de páginas o conjunto de datos.
PERFORM	Renovar o restablecer la agrupación en clúster
START	Iniciar el iniciador de canal o escuchas
STOP	Detener el iniciador de canal o escuchas

### Acciones

Las acciones que se pueden realizar para cada tipo de objeto se muestran en la tabla siguiente:

*Tabla 32. Operaciones y acciones del panel de control válidas para los objetos de IBM MQ*

Objeto	Modificar	Definir como	Gestionar (1)	Listar o Visualizar	Listar con filtro	Realizar	Iniciar	Detener
AUTHINFO	X	X	X	X	X			
CFSTATUS				X				
CFSTRUCT	X	X	X	X	X			
CHANNEL	X	X	X	X	X	X	X	X
CHSTATUS				X	X			
CLNTCONN	X	X	X	X	X			
CLUSCHL				X	X	X(2)	X(2)	X(2)
CLUSQ				X	X			
CLUSQMGR				X	X	X(2)	X(2)	X(2)
CLUSRCVR	X	X	X	X	X	X	X	X
CLUSDR	X	X	X	X	X	X	X	X
CONNECT				X	X			
CONTROL				X		X	X	X
MANAGER	X			X		X		



Tabla 32. Operaciones y acciones del panel de control válidas para los objetos de IBM MQ (continuación)

Objeto	Modificar	Definir como	Gestionar (1)	Listar o Visualizar	Listar con filtro	Realizar	Iniciar	Detener
LISTA DE NOMBRES	X	X	X	X	X			
PROCESS	X	X	X	X	X			
QALIAS	X	X	X	X	X			
QLOCAL	X	X	X	X	X			
QMODEL	X	X	X	X	X			
QREMOTE	X	X	X	X	X			
QSTATUS				X	X			
QUEUE	X	X	X	X	X			
RECEIVER	X	X	X	X	X	X	X	X
REQUESTER	X	X	X	X	X	X	X	X
seguridad	X			X		X		
SENDER	X	X	X	X	X	X	X	X
Servidor	X	X	X	X	X	X	X	X
SVRCONN	X	X	X	X	X		X	X
STGCLASS	X	X	X	X	X			
SYSTEM				X		X	X	X

**Nota:**

1. Proporciona la función Suprimir y otras funciones
2. Utilizando la acción Listar o Visualizar

 **Disposiciones de objeto en z/OS**

Puede especificar la *disposición* del objeto con el que tiene que trabajar. La disposición indica dónde se guarda la **definición** de objeto y cómo se comporta el objeto.

La disposición es importante sólo si va a trabajar con cualquiera de los siguientes tipos de objeto:

- colas
- canales
- procesos
- listas de nombres
- clases de almacenamiento
- objetos de información de autenticación

Si va a trabajar con otros tipos de objeto, la disposición no se tiene en cuenta.

Los valores permitidos son:

**Q**

QMGR. Las definiciones de objeto están en el conjunto de páginas del gestor de colas y sólo puede acceder a ellas el gestor de colas.

## C

COPY. Las definiciones de objeto están en el conjunto de páginas del gestor de colas y sólo puede acceder a ellas el gestor de colas. Son copias locales de objetos que se definen con una disposición de GROUP.

## P

PRIVATE. Las definiciones de objeto están en el conjunto de páginas del gestor de colas y sólo puede acceder a ellas el gestor de colas. Los objetos se han definido con una disposición de QMGR o COPY.

## G

GROUP. Las definiciones de objeto están en el repositorio compartido y pueden acceder a ellas todos los gestores de colas del grupo de compartición de colas.

## S

SHARED. Esta disposición sólo es aplicable a las colas locales. Las definiciones de cola están en el repositorio compartido y pueden acceder a ellas todos los gestores de colas del grupo de compartición de colas.

## A

TODOS. Si el gestor de colas de acciones es el gestor de colas de destino o \*, se incluyen objetos de **todas** las disposiciones; de lo contrario, sólo se incluyen objetos de disposiciones QMGR y COPY. Éste es el valor predeterminado.

## Selección de un gestor de colas, valores predeterminados y niveles utilizando el panel de control de ISPF en z/OS

Puede utilizar el exec CSQOREXX en ISPF para controlar los gestores de colas.

Mientras está viendo el panel inicial, no está conectado a ningún gestor de colas. Sin embargo, en cuanto pulsa Intro, se conecta al gestor de colas, o a un gestor de colas del grupo de compartición de colas especificado en el campo **Nombre de conexión**. Puede dejar este campo en blanco; esto significa que va a utilizar el gestor de colas predeterminado para aplicaciones por lotes. Esto se define en CSQBDEFV (consulte [Tarea 19: Configurar adaptadores de lotes, TSO y RRS](#) para obtener más información al respecto).

Utilice el campo **Gestor de colas de destino** para especificar el gestor de colas en el que se van a realizar las acciones que solicite. Si deja este campo en blanco, se utiliza de forma predeterminada el gestor de colas especificado en el campo **Nombre de conexión**. Puede especificar un gestor de colas de destino que no sea el gestor de colas al que se conecta. En este caso, normalmente especificaría el nombre de un objeto de gestor de colas remoto que proporciona una definición de alias de gestor de colas (el nombre se utiliza como el *NombGestColasObj* al abrir la cola de entrada de mandatos). Para ello, debe tener colas y canales adecuados configurados para acceder al gestor de colas remoto.

El campo **Gestor de colas de acciones** permite especificar que un gestor de colas que está en el mismo grupo de compartición de colas que el gestor de colas especificado en el campo **Gestor de colas de destino** sea el gestor de colas en el que se van a realizar las acciones solicitadas. Si especifica \* en este campo, las acciones que solicite se realizarán en todos los gestores de colas del grupo de compartición de colas. Si deja este campo en blanco, se utiliza de forma predeterminada el valor especificado en el campo **Gestor de colas de destino**. El campo **Gestor de colas de acciones** corresponde al uso del modificador de mandato CMDSCOPE descrito en [Mandatos MQSC](#).

### Valores predeterminados de gestor de colas

Si deja algún campo de gestor de colas en blanco o elige conectarse a un grupo de compartición de colas, se abre una ventana secundaria al pulsar **Intro**. Esta ventana confirma los nombres de los gestores de colas que va a utilizar. Pulse **Intro** para continuar. Cuando vuelva al panel inicial después de haber realizado algunas solicitudes, encontrará campos rellenos con los nombres reales.

### Niveles de gestor de colas

Los paneles de operaciones y los paneles de control solo funcionan satisfactoriamente con gestores de colas que se ejecuten en z/OS en IBM WebSphere MQ 710 o posteriores.

Si estas condiciones no se cumplen, es probable que las acciones funcionen sólo parcialmente, de forma incorrecta o no funcionen en absoluto, y que las respuestas del gestor de colas sean reconocidas.

Si el gestor de colas de acciones no está en el nivel IBM MQ 8.0.0 o superior, algunos campos no se visualizan y algunos valores no se pueden entrar. Unos cuantos objetos y acciones no están permitidos. En estos casos, se abre una ventana secundaria que le pedirá que confirme que desea continuar.

## **Utilización de las teclas de función y de la línea de mandatos con los paneles de control ISPF en z/OS**

Para utilizar los paneles, debe utilizar las teclas de función o entrar los mandatos equivalente en el área de mandatos del panel de control de ISPF.

- Teclas de función
  - Procesar las acciones
  - “Visualizar mensajes de usuario de IBM MQ” en la página 459
  - Cancelar las acciones
  - Obtención de ayuda
- Utilizar la línea de mandatos

### **Teclas de función**

Las teclas de función tienen valores especiales para IBM MQ. (Esto significa que no puede utilizar los valores predeterminados de ISPF para las teclas de función; si ha utilizado anteriormente el mandato KEYLIST OFF ISPF en cualquier lugar, debe escribir KEYLIST ON en el área de mandatos de cualquier operación y panel de control y, a continuación, pulse Intro para habilitar los valores de IBM MQ.)

Estos valores de teclas de función se pueden visualizar en los paneles, tal como se muestra en la [Figura 39 en la página 460](#). Si los valores no se muestran, escriba PFSHOW en el área de mandatos de cualquier panel de operaciones y de control, y luego pulse **Intro**. Para eliminar la visualización de los valores, utilice el mandato PFSHOW OFF.

Los valores de las teclas de función en los paneles de operaciones y los paneles de control se ajustan a los estándares de CUA. Aunque se puede cambiar el valor de la tecla mediante los procedimientos normales de ISPF (como, por ejemplo, la utilidad **KEYLIST**), no se recomienda hacerlo.

**Nota:** El uso de los mandatos **PFSHOW** Y **KEYLIST** afecta a cualquier otra pantalla ISPF lógica que se tenga y sus valores permanecen al salir de los paneles de operaciones y de control.

### **Procesar las acciones**

Pulse **Intro** para llevar a cabo la acción solicitada en un panel. La información del panel se envía al gestor de colas para su proceso.

Cada vez que se pulsa **Intro** en los paneles, IBM MQ genera uno o más mensajes de operador. Si la operación se ha realizado satisfactoriamente, recibirá el mensaje de confirmación CSQ9022I, de lo contrario recibirá algunos mensajes de error.

### **Visualizar mensajes de usuario de IBM MQ**

Pulse la tecla de función F10 en cualquier panel para ver los mensajes de usuario de IBM MQ.

### **Cancelar las acciones**

En el panel inicial, las teclas de función F3 y F12 salen de los paneles de operaciones y los paneles de control y le devuelven a ISPF. No se envía ninguna información al gestor de colas.

En cualquier otro panel, pulse las teclas de función F3 o F12 para salir del panel actual **ignorando todos los datos que ha escrito desde la última vez que pulsó Intro**. De nuevo, no se envía ninguna información al gestor de colas.

- F3 le devuelve directamente al panel inicial.

- F12 le devuelve al panel anterior.

## Obtener ayuda

Cada panel tiene paneles de ayuda asociados. Los paneles de ayuda utilizan los protocolos ISPF:

- Pulse la tecla de función F1 en cualquier panel para ver la ayuda general (ayuda ampliada) sobre la tarea.
- Pulse la tecla de función F1 con el cursor en cualquier campo para ver ayuda específica sobre ese campo.
- Pulse la tecla de función F5 desde cualquier panel de ayuda de campo para obtener la ayuda general.
- Pulse la tecla de función F3 para volver al panel base, es decir, el panel desde el que pulsó la tecla de función F1.
- Pulse la tecla de función F6 desde cualquier panel de ayuda para obtener ayuda sobre las teclas de función.

Si la información de ayuda continúa en una segunda página o en páginas subsiguientes, se visualiza un indicador **Más** en la parte superior derecha del panel. Utilice estas teclas de función para navegar por las páginas de ayuda:

- F11 para ir a la página de ayuda siguiente (si hay una).
- F10 para volver a la página de ayuda anterior (si hay una).

## Utilización de la línea de mandatos

No necesita utilizar la línea de mandatos para emitir los mandatos utilizados por los paneles de operaciones y los paneles de control, ya que están disponibles desde las teclas de función. La línea de mandatos se proporciona para poder ejecutar mandatos ISPF normales (como **PFSHOW**).

El mandato ISPF PANELID ON muestra el nombre del panel CSQOREXX actual.

La línea de mandatos se visualiza inicialmente en la posición predeterminada situada en la parte inferior de los paneles, independientemente de qué valores ISPF tenga. Se puede utilizar el mandato SETTINGS ISPF desde cualquiera de las operaciones y paneles de control para cambiar la posición de la línea de mandatos. Los valores se recuerdan para sesiones posteriores con los paneles de operaciones y los paneles de control.

## Uso de los paneles de control y operaciones en z/OS

Utilice este tema para investigar el panel de control inicial visualizado desde CSQOREXX

La [Figura 39](#) en la [página 460](#) muestra el panel que se visualiza al iniciar una sesión de panel.

```

IBM MQ for z/OS - Main Menu
Complete fields. Then press Enter.
Action . . . . . 1      0. List with filter   4. Manage
                   1. List or Display   5. Perform
                   2. Define like     6. Start
                   3. Alter         7. Stop
                   8. Command
Object type . . . . . CHANNEL      +
Name . . . . . *
Disposition . . . . . A  Q=Qmgr, C=Copy, P=Private, G=Group,
                       S=Shared, A=All

Connect name . . . . . MQ1C - local queue manager or group
Target queue manager . . . . . MQ1C
- connected or remote queue manager for command input
Action queue manager . . . . . MQ1C - command scope in group
Response wait time . . . . . 30 5 - 999 seconds

(C) Copyright IBM Corporation 1993, 2024. All rights reserved.

Command ==>
F1=Help      F2=Split   F3=Exit     F4=Prompt   F9=SwapNext F10=Messages
F12=Cancel

```

Figura 39. El panel inicial de operaciones y de control de IBM MQ

Desde este panel puede realizar acciones tales como:

- Elegir el gestor de colas local que desea y si desea que los mandatos se emitan en ese gestor de colas, en un gestor de colas remoto o en otro gestor de colas del mismo grupo de compartición de colas que el gestor de colas local. Escriba encima del nombre del gestor de colas si necesita cambiarlo.
- Seleccionar la acción que desea realizar escribiendo el número correspondiente en el campo **Acción**.
- Especifique el tipo de objeto con el que desea trabajar. Pulse la tecla de función F1 para obtener ayuda sobre los tipos de objeto si no está seguro de cuáles son.
- Especificar la disposición del tipo de objeto con el que desea trabajar.
- Visualizar una lista de objetos del tipo especificado. Escriba un asterisco (\*) en el campo **Nombre** y pulse **Intro** para visualizar una lista de objetos (del tipo especificado) ya definidos en el gestor de colas de acciones. A continuación puede seleccionar uno o más objetos con los que trabajar en secuencia. Todas las acciones están disponibles en la lista.

**Nota:** Se recomienda que haga elecciones que den lugar a que se visualice una lista de objetos, y luego trabaje a partir de esa lista. Utilice la acción **Visualizar**, ya que está permitida para todos los tipos de objeto.

## **Uso del recurso de mandatos en z/OS**

Utilice el editor para entrar o modificar los mandatos MQSC que se van a pasar al gestor de colas.

Desde el panel principal, CSQOPRIA, seleccione la opción **8 Mandato**, para iniciar el Recurso de mandatos.

Aparecerá una sesión de edición de un archivo secuencial, *prefijo*.CSQUTIL.COMMANDS, que se utiliza como entrada para la función COMMAND de CSQUTIL; consulte [Emisión de mandatos a IBM MQ](#).

No es necesario prefijar los mandatos con la serie de prefijo de mandato (CPF).

Puede continuar los mandatos MQSC en las líneas subsiguientes terminando la línea actual con los caracteres de continuación + o -. De forma alternativa, utilice la modalidad de edición de línea para proporcionar mandatos MQSC largos o los valores de atributo largos dentro del mandato.

### **edición de línea**

Para utilizar la edición de línea, mueva el cursor a la línea correspondiente en el panel de edición y utilice **F4** para visualizar una sola línea en un panel desplazable. Una sola línea puede tener hasta 32760 bytes de datos.

Para salir de la edición de línea:

- **F3 salir** guarda los cambios realizados en la línea y sale de la edición de línea
- **F12 cancelar** vuelve al panel de edición descartando los cambios realizados en la línea.

Para descartar los cambios realizados en la sesión de edición, utilice **F12 cancelar** para terminar la sesión de edición sin cambiar el contenido del archivo. Los mandatos no se ejecutan.

### **Ejecución de mandatos**

Cuando haya terminado de entrar mandatos MQSC, termine la sesión de edición con **F3 salir** para guardar el contenido del archivo e invocar CSQUTIL para pasar los mandatos al gestor de colas. La salida del proceso del mandato se guarda en el archivo *prefijo*.CSQUTIL.OUTPUT. Se abre automáticamente una sesión de edición en este archivo para que pueda ver las respuestas. Pulse **F3 salir** para salir de esta sesión y volver al menú principal.

## **Cómo trabajar con objetos IBM MQ en z/OS**

Muchas de las tareas descritas en esta documentación requieren la manipulación de objetos de IBM MQ. Los tipos de objeto son gestores de colas, colas, definiciones de proceso, listas de nombres, canales, canales de conexión de cliente, escuchas, servicios y objetos de información de autenticación.

- [Definición de objetos de cola simples](#)

- [Definición de otros tipos de objetos](#)
- [Trabajar con definiciones de objeto](#)
- [Trabajar con listas de nombres](#)

## Definición de objetos de cola simples

Para definir un nuevo objeto, utilice una definición existente como base para ello. Puede hacer esto de una de tres maneras:

- Seleccionando un objeto que sea miembro de una lista visualizada como resultado de las opciones seleccionadas en el panel inicial. A continuación, entre el tipo de acción 2 (**Definir como**) en el campo de acción situado junto al objeto seleccionado. El nuevo objeto tiene los atributos del objeto seleccionado, excepto la disposición. Luego, puede cambiar los atributos de su nuevo objeto como desee.
- En el panel inicial, seleccione el tipo de acción **Definir como**, entre el tipo de objeto que está definiendo en el campo **Tipo de objeto** y escriba el nombre de un objeto específico existente en el campo **Nombre**. El nuevo objeto tiene los mismos atributos que el objeto que ha especificado en el campo **Nombre**, excepto la disposición. Luego, puede cambiar los atributos de su nueva definición de objeto como desee.
- Seleccionando el tipo de acción **Definir como**, especificando un tipo de objeto y luego dejando el campo **Nombre** en blanco. A continuación, puede definir el nuevo objeto, que tiene los atributos predeterminados definidos para su instalación. Luego, puede cambiar los atributos de su nueva definición de objeto como desee.

**Nota:** No especifique el nombre del objeto que está definiendo en el panel inicial, sino en el panel **Definir** que aparece.

El ejemplo siguiente muestra cómo definir una cola local utilizando una cola existente como plantilla.

### Definición de una cola local

Para definir un objeto de cola local desde los paneles de operaciones y los paneles de control, utilice una definición de cola existente como base para la nueva definición. Hay varios paneles que completar. Cuando haya completado todos los paneles y esté convencido de que los atributos son correctos, pulse Intro para enviar la definición al gestor de colas, que crea entonces la cola real.

Utilice la acción **Definir como** bien en el panel inicial o para una entrada de objeto de una lista que se visualiza como resultado de las opciones seleccionadas en el panel inicial.

Por ejemplo, empezando desde el panel inicial, rellene estos campos:

<b>Acción</b>	2 (Definir como)
<b>Tipo de objeto</b>	QLOCAL
<b>Nombre</b>	QUEUE.YOU.LIKE. Este es el nombre de la cola que proporciona los atributos para la nueva cola.

Pulse Intro para visualizar el panel **Definir una cola local**. El campo de nombre de cola está en blanco para que pueda proporcionar el nombre para la nueva cola. La descripción es la de la cola en la que está basando esta nueva definición. Sobrescriba este campo con su propia descripción para la nueva cola.

Los valores en los otros campos son los de la cola en la que está basando esta nueva cola, excepto la disposición. Puede sobrescribir estos campos como sea necesario. Por ejemplo, escriba Y en el campo **Transferencia habilitada** (si no es ya Y) si las aplicaciones debidamente autorizadas pueden transferir mensajes a esta cola.

Puede obtener ayuda de campo colocando el cursor en un campo y pulsando la tecla de función F1. La ayuda de campo proporciona información sobre los valores que se pueden utilizar para cada atributo.

Cuando haya completado el primer panel, pulse la tecla de función F8 para visualizar el segundo panel.

#### **Sugerencias:**

1. No pulse Intro en esta etapa, de lo contrario, se creará la cola antes de que tenga la oportunidad de completar los campos restantes. (Si pulsa Intro antes de tiempo, no se preocupe; siempre puede modificar la definición más adelante.)
2. No pulse las teclas de función F3 o F12 o los datos que ha entrado se perderán.

Pulse la tecla de función F8 repetidamente para ver y completar los paneles restantes, incluyendo los paneles de definición de desencadenador, de control de sucesos y de informes de restitución.

#### **Cuando haya completado la definición de la cola local**

Cuando haya completado la definición, pulse Intro para enviar la información al gestor de colas para su proceso. El gestor de colas crea la cola de acuerdo con la definición que ha proporcionado. Si no desea crear la cola, pulse la tecla de función F3 para salir y cancelar la definición.

### **Definición de otros tipos de objetos**

Para definir otros tipos de objeto, utilice una definición existente como base para la nueva definición, tal como se explica en [Definir una cola local](#).

Utilice la acción **Definir como** bien en el panel inicial o para una entrada de objeto de una lista que se visualiza como resultado de las opciones seleccionadas en el panel inicial.

Por ejemplo, empezando desde el panel inicial, rellene estos campos:

<b>Acción</b>	2 (Definir como)
<b>Tipo de objeto</b>	QALIAS, NAMELIST, PROCESS, CANAL, y otros objetos de recurso.
<b>Nombre</b>	Déjelo en blanco o escriba el nombre de un objeto existente del mismo tipo.

Pulse Intro para visualizar los paneles DEFINE correspondientes. Rellene los campos según sea necesario y luego pulse de nuevo Intro para enviar la información al gestor de colas.

Al igual que con la definición de una cola local, la definición de otro tipo de objeto requiere la cumplimentación de varios paneles. La definición de una lista de nombres requiere un trabajo adicional, tal como se describe en [“Trabajar con listas de nombres”](#) en la página 464.

### **Trabajar con definiciones de objeto**

Cuando se ha definido un objeto, puede especificar una acción en el campo **Acción** para modificar, visualizar o gestionarlo.

En cada caso, puede:

- Seleccionar el objeto con el que desea trabajar en una lista que se visualiza como resultado de las opciones seleccionadas en el panel inicial. Por ejemplo, si ha especificado 1 en el campo **Acción** para visualizar objetos, Queue en el campo **Tipo de objeto** y \* en el campo **Nombre**, aparecerá una lista de todas las colas definidas en el sistema. Podrá entonces seleccionar en esta lista la cola con la que necesita trabajar.
- Empezar desde el panel inicial, donde especificará el objeto con el que va a trabajar completando los campos **Tipo de objeto** y **Nombre**.

#### **Modificar una definición de objeto**

Para modificar una definición de objeto, especifique la acción 3 y pulse Intro para ver los paneles ALTER. Estos paneles son muy similares a los paneles DEFINE. Puede modificar los valores que desee. Cuando finalice los cambios, pulse Intro para enviar la información al gestor de colas.

### Visualizar una definición de objeto

Si desea ver los detalles de un objeto sin la posibilidad de cambiarlos, especifique la acción 1 y pulse Intro para ver los paneles DISPLAY. De nuevo, estos paneles son similares a los paneles DEFINE, excepto que no puede cambiar ninguno de los campos. Cambie el nombre de objeto para ver los detalles de otro objeto.

### Suprimir un objeto

Para suprimir un objeto, especifique la acción 4 (Gestionar) y la acción **Suprimir** es una de las acciones que se muestran en el menú resultante. Seleccione la acción **Suprimir**.

Se le pedirá que confirme su solicitud. Si pulsa la tecla de función F3 o F12, la solicitud se cancela. Si pulsa Intro, la solicitud se confirma y se pasa al gestor de colas. El objeto que ha especificado se suprimirá.

**Nota:** No puede suprimir la mayoría de los tipos de objeto de canal a menos que el iniciador de canal esté iniciado.

### Trabajar con listas de nombres

Al trabajar con listas de nombres, proceda como lo haría con otros objetos.

Para las acciones DEFINE LIKE o ALTER, pulse la tecla de función F11 para añadir nombres a la lista o para cambiar los nombres de la lista. Esto implica trabajar con el editor ISPF y todos los mandatos de edición ISPF normales estarán disponibles. Especifique cada nombre en la lista de nombres en una línea separada.

Cuando utilice el editor ISPF de este modo, los valores de las teclas de función son los valores ISPF normales y **no** los utilizados por los otros paneles de operaciones y los paneles de control.

Si tiene que especificar nombres en minúsculas en la lista, especifique CAPS(OFF) en la línea de mandatos del panel del editor. Cuando haga esto, todas las listas de nombres que edite en el futuro estarán en minúsculas hasta que especifique CAPS(ON).

Cuando haya terminado de editar la lista de nombres, pulse la tecla de función F3 para finalizar la sesión de edición de ISPF. A continuación, pulse Intro para enviar los cambios al gestor de colas.

**Atención:** Si no pulsa Intro en este momento, sino que en su lugar pulsa la tecla de función F3, perderá todas las actualizaciones que haya escrito.

## Implementación del sistema utilizando varias colas de transmisión del clúster

No existe ninguna diferencia si el canal se utiliza en un solo clúster o en un clúster solapante. Cuando se selecciona e inicia el canal, éste selecciona la cola de transmisión en función de las definiciones.

### Procedimiento

- Si está utilizando la opción DEFCLXQ, consulte [“Utilización de la definición automática de las colas y la conmutación”](#) en la página 464.
- Si está utilizando un método por etapas, consulte [“Cambio de los canales de clúster emisor mediante un método por etapas”](#) en la página 465.

## Utilización de la definición automática de las colas y la conmutación

Utilice esta opción si tiene pensado utilizar la opción DEFCLXQ. Se creará una cola para cada canal y cada nuevo canal.

### Procedimiento

1. Revise la definición de SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE y cambie los atributos, si es necesario.



Esta cola está definida en el miembro SCSQPROC (csq4insx).

2. Cree la cola modelo SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE.
3. Aplique políticas de seguridad para esta cola modelo y SYSTEM.CLUSTER.TRANSMIT. \* \* colas.  
Para z/OS, el ID de usuario de tarea iniciada del iniciador de canal necesita:

- Acceso de control a CLASS(MQADMIN) para

```
ssid.CONTEXT.SYSTEM.CLUSTER.TRANSMIT.channelname
```

- Acceso de actualización a CLASS(MQQUEUE) para

```
ssid.SYSTEM.CLUSTER.TRANSMIT.channelname
```

### **Cambio de los canales de clúster emisor mediante un método por etapas**

Utilice esta opción si tiene previsto usar un enfoque por fases. Este proceso le permite pasar a los nuevos canales de clúster-emisor varias veces para atenderse a las necesidades de su empresa.

#### **Antes de empezar**

- Identifique las aplicaciones empresariales y los canales que se utilizan.
- En el caso de las colas que utiliza, visualice los clústeres en las que están ubicadas.
- Visualice los canales para mostrar los nombres de conexión, los nombres de los gestores de colas remotos y a qué clústeres da soporte el canal.

#### **Acerca de esta tarea**

- Cree una cola de transmisión. En z/OS, puede ser conveniente considerar qué conjunto de páginas desea utilizar para la cola.
- Configure la política de seguridad para la cola.
- Cambie cualquier supervisión de cola para incluir el nombre de cola.
- Decida qué canales han de utilizar esta cola de transmisión. Los canales deben tener un nombre similar, por lo tanto, los caracteres genéricos '\*' de CLCHNAME identifican el canal.
- Cuando esté preparado para utilizar la nueva función, altere la cola de transmisión para especificar el nombre de los canales que han de utilizar esta cola de transmisión. Por ejemplo, CLUSTER1.TOPARISO CLUSTER1.\* o \*.TOPARIS
- Inicie los canales

#### **Procedimiento**

1. Utilice el mandato DIS CLUSQMGR (xxxx) XMITQ para visualizar los canales emisores del clúster definidos en el clúster, donde xxxx es el nombre del gestor de colas remoto.
2. Configure el perfil de seguridad para la cola de transmisión y conceda a la cola acceso al iniciador de canal.
3. Defina la cola de transmisión que se ha de utilizar y especifique USAGE(XMITQ) INDXTYPE( CORRELID ) SHARE y CLCHNAME(valor)

El ID de usuario de tarea iniciada del iniciador de canal necesita el acceso siguiente:

```
alter class(MQADMIN) ssid.CONTEXT.SYSTEM.CLUSTER.TRANSMIT.channel  
update class(MQQUEUE ssid.SYSTEM.CLUSTER.TRANSMIT.channel
```

y el Id de usuario que utiliza el mandato SWITCH necesita el acceso siguiente:

```
alter cl(MQADMIN) ssid.QUEUE.queuname
```

#### 4. Detenga y reinicie los canales.

El cambio de canal se lleva a cabo cuando se inicia el canal mediante un mandato MQSC o cuando utiliza CSQUTIL. Puede identificar qué canales se han de reiniciar utilizando SWITCH CHANNEL (\*) STATUS de CSQUTIL

Si tiene problemas durante el inicio del canal, detenga el canal, resuelva los problemas y reinicie el canal.

Tenga en cuenta que puede cambiar el atributo CLCHNAME tantas veces como lo necesite.

El valor de CLCHNAME utilizado es el valor cuando se inicia el canal, por lo tanto, puede cambiar la definición de CLCHNAME mientras el canal continúa utilizando las definiciones del momento en que se ha iniciado el canal. El canal utiliza la nueva definición cuando se reinicia.

## **Deshacer un cambio en z/OS**

Necesita tener un proceso para restituir un cambio si los resultados no son los previstos.

### ¿Qué puede salir mal?

Si la nueva cola de transmisión no es la que esperaba:

1. Compruebe si el CLCHNAME es tal como lo esperaba
2. Revise el registro de trabajo para comprobar si ha finalizado el proceso de conmutación. Si no es así, espere y compruebe la nueva cola de transmisión del canal más tarde.

Si está utilizando varias colas de transmisión, es importante que diseñe definiciones de colas de transmisión de forma explícita y evite una configuración solapante complicada. De este modo, puede asegurarse de que si existen problemas, puede regresar a las colas y a la configuración originales.

Si encuentra problemas cuando pasa a utilizar una cola de transmisión diferente, debe resolver los problemas antes de continuar con el cambio.

Se debe completar una solicitud de cambio existente antes de poder realizar una solicitud de cambio nueva. Por ejemplo, si:

1. Define una cola de transmisión nueva con una profundidad máxima de uno y hay 10 mensajes que esperan ser enviados.
2. Cambia la cola de transmisión para especificar el nombre de canal en el parámetro CLCHNAME.
3. Detiene y reinicia el canal. El intento de mover los mensajes falla y se informa acerca de problemas.
4. Cambia el parámetro CLCHNAME en la cola de transmisión para que esté en blanco.
5. Detiene y reinicia el canal. El canal continúa intentando la operación y completa la solicitud original, por lo tanto, el canal continúa utilizando la nueva cola de transmisión.
6. Es necesario que resuelva los problemas y reinicie el canal, de modo que el traslado de los mensajes se realice correctamente.

La próxima vez que se reinicia el canal captura cualquier cambio, de modo que si ha establecido CLCHNAME en blancos, el canal no utilizará la cola de transmisión especificado.

En este ejemplo, el que se haya cambiado CLCHNAME en la cola de transmisión a blancos no necesariamente significa que el canal utilice la cola SYSTEM.CLUSTER.TRANSMIT, ya que pueden haber otras colas de transmisión cuyo parámetro CLCHNAME coincida con el nombre de canal. Por ejemplo, se puede haber definido un nombre genérico o es posible que el atributo DEFCLXQ se haya establecido en el canal, de modo que el canal utiliza la cola dinámica, en lugar de la cola SYSTEM.CLUSTER.TRANSMIT.

## Escribir programas para administrar IBM MQ for z/OS

Puede escribir sus propios programas de aplicación para administrar un gestor de colas. Utilice este tema para comprender los requisitos para escribir sus propios programas de administración.

### Inicio de la información de interfaz de programación de uso general

Este conjunto de temas contiene sugerencias y recomendaciones para permitirle emitir mandatos de IBM MQ desde un programa de IBM MQ.

**Nota:** En este tema, las llamadas de MQI se describen utilizando la notación del lenguaje C. Para las invocaciones típicas de las llamadas en los lenguajes COBOL, PL/I y de ensamblador, consulte el manual [Llamadas a función](#).

### Compresión de cómo funciona todo

En resumen, el procedimiento para emitir mandatos desde un programa de aplicación es el siguiente:

1. Cree un mandato de IBM MQ en un tipo de mensaje de IBM MQ llamado un *mensaje de solicitud*. El mandato puede tener el formato MQSC o PCF.
2. Envíe (utilice MQPUT) este mensaje a una cola especial llamada la cola de entrada del mandato del sistema. El procesador de mandatos de IBM MQ ejecuta el mandato.
3. Recupere (utilice MQGET) los resultados del mandato como *mensajes de respuesta* en la cola de respuesta. Estos mensajes contienen los mensajes de usuario que deben determinar si el mandato se ha realizado correctamente y, en ese caso, cuáles han sido los resultados.

A continuación, es responsabilidad del programa de aplicación procesar los resultados.

Este conjunto de temas contiene:

## Preparación de colas para los programas de administración

Los programas de administración requieren una serie de colas predefinidas para la entrada de mandatos del sistema y para recibir respuestas.

Esta sección se aplica a mandatos en el formato MQSC. Para obtener el equivalente en PCF, consulte [“Utilización de los formatos de mandato programable de IBM MQ” en la página 26](#).

Antes de emitir cualquier llamada MQPUT o MQGET, primero debe definir, y luego abrir, las colas que va a utilizar.

### Definir la cola de entrada de mandatos del sistema

La cola de entrada de mandatos del sistema es una cola local llamada SYSTEM.COMMAND.INPUT. El conjunto de datos de inicialización CSQINP2 suministrado, thlqual.SCSQPROC(CSQ4INSG), contiene una definición predeterminada para la cola de entrada de mandatos del sistema. Por compatibilidad con IBM MQ en otras plataformas, un alias de esta cola, denominado SYSTEM.ADMIN.COMMAND.QUEUE. Consulte [Definiciones de ejemplo suministradas con IBM MQ](#) para obtener más información.

### Definir una cola de respuesta

Debe definir una cola de respuesta para recibir mensajes de respuesta del procesador de mandatos de IBM MQ. Puede ser cualquier cola con atributos que permitan transferir mensajes de respuesta a la misma. Sin embargo, para el funcionamiento normal, especifique estos atributos:

- USAGE(NORMAL)
- NOTRIGGER (a menos que su aplicación utilice desencadenamiento)

Evite utilizar mensajes persistentes para los mandatos, pero si decide hacerlo, la cola de respuesta no debe ser una cola dinámica temporal.

El conjunto de datos de inicialización CSQINP2 suministrado, thlqual.SCSQPROC(CSQ4INSG), contiene una definición para una cola modelo llamada SYSTEM.COMMAND.REPLY.MODEL. Puede utilizar este modelo para crear una cola de respuesta dinámica.

**Nota:** Las respuestas generadas por el procesador de mandatos pueden tener hasta 15000 bytes de longitud.

Si utiliza una cola dinámica permanente como cola de respuesta, la aplicación debe dar tiempo a que se completen todas las operaciones PUT y GET antes de intentar suprimir la cola, de lo contrario se puede devolver MQRC2055 (MQRC\_Q\_NOT\_EMPTY). Si esto ocurre, intente de nuevo la supresión de cola al cabo de unos segundos.

### **Abrir la cola de entrada de mandatos del sistema**

Para poder abrir la cola de entrada de mandatos del sistema, el programa de aplicación debe estar conectado al gestor de colas. Utilice la llamada MQI MQCONN o MQCONNX para hacerlo.

A continuación, utilice la llamada MQI MQOPEN para abrir la cola de entrada de mandatos del sistema. Para utilizar esta llamada:

1. Establezca el parámetro **Options** en MQOO\_OUTPUT
2. Establezca los campos de descriptor de objeto MQOD de la siguiente manera:

**ObjectType**

MQOT\_Q (el objeto es una cola)

**ObjectName**

SYSTEM.COMMAND.INPUT

**ObjectQMgrName**

Si desea enviar los mensajes de solicitud a su gestor de colas local, deje este campo en blanco. Esto significa que los mandatos se procesan localmente.

Si desea que los mandatos de IBM MQ se procesen en un gestor de colas remoto, especifique el nombre aquí. También debe tener las colas y los enlaces correctos establecidos, tal como se describe en [Transferencia a colas distribuidas y clústeres](#).

### **Abrir una cola de respuesta**

Para recuperar las respuestas de un mandato de IBM MQ, debe abrir una cola de respuesta. Una forma de hacerlo es especificar la cola modelo, SYSTEM.COMMAND.REPLY.MODEL, en una llamada MQOPEN, para crear una cola dinámica permanente como la cola de respuesta. Para utilizar esta llamada:

1. Establezca el parámetro **Options** en MQOO\_INPUT\_SHARED
2. Establezca los campos de descriptor de objeto MQOD de la siguiente manera:

**ObjectType**

MQOT\_Q (el objeto es una cola)

**ObjectName**

El nombre de la cola de respuesta. Si el nombre de cola que se especifica es el nombre de un objeto de cola modelo, el gestor de colas crea una cola dinámica.

**ObjectQMgrName**

Para recibir las respuestas en su gestor de colas local, deje este campo en blanco.

**DynamicQName**

Especifique el nombre de la cola dinámica que se va a crear.

El servidor de mandatos es un componente de IBM MQ que funciona con el componente procesador de mandatos. Puede enviar mensajes con formato al servidor de mandatos, que interpreta los mensajes, ejecuta las solicitudes de administración y envía respuestas a la aplicación de administración.

El servidor de mandatos lee los mensajes de solicitud de la cola de entrada de mandatos del sistema, los verifica y pasa los que son válidos como mandatos al procesador de mandatos. El procesador de mandatos procesa los mandatos y transfiere las respuestas como mensajes de respuesta a la cola de respuesta que especifique. El primer mensaje de respuesta contiene el mensaje de usuario CSQN205I. Consulte [“Interpretar los mensajes de respuesta del servidor de mandatos”](#) en la página 473 para obtener más información. El servidor de mandatos también procesa mandatos de iniciador de canal y de grupo de compartición de colas desde dondequiera que se emitan.

### **Identificar el gestor de colas que procesa los mandatos**

El gestor de colas que procesa los mandatos que se emiten desde un programa de administración es el gestor de colas propietario de la cola de entrada de mandatos del sistema a la que se transfiere el mensaje.

### **Iniciar el servidor de mandatos**

Normalmente, el servidor de mandatos se inicia automáticamente cuando se inicia el gestor de colas. Queda disponible en cuanto el mandato START QMGR devuelve el mensaje CSQ9022I 'START QMGR' NORMAL COMPLETION. El servidor de mandatos se detiene cuando todas las tareas relacionadas se han desconectado durante la fase de terminación del sistema.

Puede controlar usted mismo el servidor de mandatos mediante los mandatos START CMDSERV y STOP CMDSERV. Para impedir que el servidor de mandatos se inicie automáticamente cuando se reinicie IBM MQ, puede añadir un mandato STOP CMDSERV a los conjuntos de datos de inicialización CSQINP1 o CSQINP2. No obstante, esto no se recomienda ya que impide que se procese cualquier mandato de iniciador de canal o de grupo de compartición de colas.

El mandato STOP CMDSERV detiene el servidor de mandatos en cuanto éste ha acabado de procesar el mensaje actual, o inmediatamente si no se está procesando ningún mensaje.

Si el servidor de mandatos se ha detenido mediante un mandato STOP CMDSERV en el programa, no se puede procesar ningún otro mandato del programa. Para reiniciar el servidor de mandatos, debe emitir un mandato START CMDSERV desde la consola de z/OS.

Si detiene y reinicia el servidor de mandatos mientras el gestor de colas está en ejecución, todos los mensajes que están en la cola de entrada de mandatos del sistema cuando el servidor de mandatos se detiene se procesan cuando se reinicia el servidor de mandatos. Sin embargo, si detiene y reinicia el gestor de colas después de detener el servidor de mandatos, sólo los mensajes persistentes de la cola de entrada de mandatos del sistema se procesan cuando se reinicia el servidor de mandatos. Todos los mensajes no persistentes de la cola de entrada de mandatos del sistema se pierden.

### **Enviar mandatos al servidor de mandatos**

Para cada mandato, cree un mensaje que contenga el mandato y luego colóquelo en la cola de entrada de mandatos del sistema.

### **Creación de un mensaje que incluya los mandatos de IBM MQ**

Puede incorporar mandatos de IBM MQ en un programa de aplicación mediante la creación de mensajes de solicitud que incluyan los mandatos necesarios. Para cada uno de estos mandatos:

1. Cree un almacenamiento intermedio que contenga una serie de caracteres que represente el mandato.
2. Emita una llamada MQPUT especificando el nombre del almacenamiento intermedio en el parámetro **buffer** de la llamada.

La forma más sencilla de hacer esto en C es definir un almacenamiento intermedio utilizando 'char'.  
Por ejemplo:

```
char message_buffer[ ] = "ALTER QLOCAL(SALES) PUT(ENABLED)";
```

Cuando cree un mandato, utilice una serie de caracteres terminada en nulo. No especifique una serie de prefijo de mandato (CPF) al comienzo de un mandato definido de esta manera. Esto significa que no necesita modificar los scripts de mandatos si desea ejecutarlos en otro gestor de colas. No obstante, debe tener en cuenta que se incluye un CPF en todos los mensajes de respuesta que se transfieren a la cola de respuesta.

El servidor de mandatos convierte todos los caracteres en minúsculas a mayúsculas a menos que estén entre comillas.

Los mandatos puede tener cualquier longitud hasta un máximo 32762 caracteres.

### Colocar mensajes en la cola de entrada de mandatos del sistema

Utilice la llamada MQPUT para colocar mensajes de petición que contienen mandatos en la cola de entrada de mandatos del sistema. En esta llamada, debe especificar el nombre de la cola de respuesta que ya ha abierto.

Para utilizar la llamada MQPUT:

1. Configure estos parámetros MQPUT:

**Hconn**

El manejador de conexión devuelto por la llamada MQCONN o MQCONNX.

**Hobj**

El manejador de objeto devuelto por la llamada MQOPEN para la cola de entrada de mandatos del sistema.

**BufferLength**

La longitud del mandato con formato.

**Buffer**

El nombre del almacenamiento intermedio que contiene el mandato.

2. Establezca estos campos de MQMD:

**MsgType**

MQMT\_REQUEST

**Format**

MQFMT\_STRING o MQFMT\_NONE

Si no utiliza la misma página de códigos que el gestor de colas, establezca *CodedCharSetId* según corresponda y establezca MQFMT\_STRING, para que el servidor de mandatos puede convertir el mensaje. No establezca MQFMT\_ADMIN, ya que esto hace que el mandato sea interpretado como PCF.

**ReplyToQ**

Nombre de la cola de respuesta.

**ReplyToQMgr**

Si desea que las respuestas se envíen a su gestor de colas local, deje este campo en blanco. Si desea que los mandatos de IBM MQ se envíen a un gestor de colas remoto, especifique el nombre aquí. También debe tener las colas y los enlaces correctos establecidos, tal como se describe en [Transferencia a colas distribuidas y clústeres](#).

3. Establezca otros campos de MQMD, según sea necesario. Normalmente debería utilizar mensajes no persistentes para los mandatos.
4. Establezca las opciones *PutMsgOpts*, según sea necesario.

Si especifica MQPMO\_SYNCPOINT (el valor predeterminado), debe seguir la llamada MQPUT con una llamada de punto de sincronización.

### Utilización de MQPUT1 y de la cola de entrada de mandatos del sistema

Si desea colocar un solo mensaje en la cola de entrada de mandatos del sistema, puede utilizar la llamada **MQPUT1**. Esta llamada combina las funciones de una **MQOPEN**, seguida de una **MQPUT** de un mensaje, seguida de una **MQCLOSE**, todas ellas en una sola llamada. Si utiliza esta llamada, modifique los parámetros en consecuencia. Consulte [Transferencia de un mensaje a una cola utilizando la llamada MQPUT1](#) para obtener más información.

## Recuperar las respuestas a los mandatos

El servidor de mandatos envía una respuesta a una cola de respuesta para cada mensaje de solicitud que recibe. Cualquier aplicación de administración debe recibir y manejar los mensajes de respuesta.

Cuando el procesador de mandatos procesa los mandatos, los mensajes de respuesta se transfieren a la cola de respuesta especificada en la llamada MQPUT. El servidor de mandatos envía los mensajes de respuesta con la misma persistencia que el mensaje de mandato que ha recibido.

### Esperar una respuesta

Utilice la llamada MQGET para recuperar una respuesta de su mensaje de solicitud. Un mensaje de solicitud puede producir varios mensajes de respuesta. Para obtener más información, consulte [“Interpretar los mensajes de respuesta del servidor de mandatos”](#) en la página 473.

Puede especificar un intervalo de tiempo durante el cual una llamada MQGET espera a que se genere un mensaje de respuesta. Si no obtiene una respuesta, utilice la lista de comprobación que empieza en el tema [“Si no recibe una respuesta”](#) en la página 473.

Para utilizar la llamada MQGET:

1. Establezca estos parámetros:

#### **Hconn**

El manejador de conexión devuelto por la llamada MQCONN o MQCONNX.

#### **Hobj**

El manejador de objeto devuelto por la llamada MQOPEN para la cola de respuesta.

#### **Buffer**

El nombre del área para recibir la respuesta.

#### **BufferLength**

La longitud del almacenamiento intermedio para recibir la respuesta. Esta debe ser como mínimo de 80 bytes.

2. Para asegurarse de que sólo recibe las respuestas del mandato que ha emitido, debe especificar los campos *MsgId* y *CorrelId* adecuados. Estos dependen de las opciones de informe, MQMD\_REPORT, que ha especificado en la llamada MQPUT:

#### **MQRO\_NONE**

Cero binario, '00...00' (24 nulos).

#### **MQRO\_NEW\_MSG\_ID**

Cero binario, '00...00' (24 nulos).

Este es el valor predeterminado si no se ha especificado ninguna de estas opciones.

#### **MQRO\_PASS\_MSG\_ID**

El *MsgId* de la llamada MQPUT.

#### **MQRO\_NONE**

El *MsgId* de la llamada MQPUT.

**MQRO\_COPY\_MSG\_ID\_TO\_CORREL\_ID**

El *MsgId* de la llamada MQPUT.

Este es el valor predeterminado si no se ha especificado ninguna de estas opciones.

**MQRO\_PASS\_CORREL\_ID**

El *CorrelId* de la llamada MQPUT.

Para obtener más detalles sobre las opciones de informes, consulte [Opciones de informes y distintivos de mensajes](#).

3. Establezca los siguientes campos de *GetMsgOpts*:

**Options**

MQGMO\_WAIT

Si no utiliza la misma página de códigos que el gestor de colas, establezca MQGMO\_CONVERT y establezca *CodedCharSetId* según corresponda en el MQMD.

**WaitInterval**

Para las respuestas del gestor de colas local, pruebe con 5 segundos. Codificado en milisegundos, esto se convierte en 5000. Para las respuestas de un gestor de colas remoto, y mandatos de estado y de control de canal, pruebe con 30 segundos. Codificado en milisegundos, esto se convierte en 30000.

**Mensajes descartados**

Si el servidor de mandatos detecta que un mensaje de solicitud no es válido, descarta este mensaje y graba el mensaje *CSQN205I* en la cola de respuesta especificada. Si no hay ninguna cola de respuesta, el mensaje *CSQN205I* se transfiere a la cola de mensajes no entregados. El código de retorno en este mensaje indica por qué el mensaje de solicitud original no era válido:

- 00D5020F** No es de tipo MQMT\_REQUEST.
- 00D50210** Tiene longitud cero.
- 00D50212** Tiene una longitud superior a 32762 bytes.
- 00D50211** Sólo contiene espacios en blancos.
- 00D5483E** Necesitaba conversión, pero *Format* no era MQFMT\_STRING.
- Otros** Consulte [Códigos del servidor de mandatos](#)

**El descriptor de mensaje de respuesta del servidor de mandatos**

Para cualquier mensaje de respuesta, se establecen los siguientes campos de descriptor de mensaje MQMD:

- MsgType* MQMT\_REPLY
- Feedback* MQFB\_NONE
- Encoding* MQENC\_NATIVE
- Priority* Igual que para el MQMD en el mensaje que ha emitido.
- Persistence* Igual que para el MQMD en el mensaje que ha emitido.
- CorrelId* Depende de las opciones de informe de MQPUT.
- ReplyToQ* Ninguno.

El servidor de mandatos establece el campo *Options* de la estructura MQPMO en MQPMO\_NO\_SYNCPOINT. Esto significa que puede recuperar las respuestas a medida que se crean, en lugar de recuperarlas como un grupo en el siguiente punto de sincronización.



## Interpretar los mensajes de respuesta del servidor de mandatos

Cada mensaje de solicitud procesado correctamente por IBM MQ genera al menos dos mensajes de respuesta. Cada mensaje de respuesta contiene un solo mensaje de usuario de IBM MQ.

La longitud de una respuesta depende del mandato que se ha emitido. La respuesta más larga que se puede obtener es la de un mandato DISPLAY NAMELIST, y está puede tener hasta 15000 bytes de longitud.

El primer mensaje de usuario, CSQN205I, siempre contiene:

- Un recuento de las respuestas (en decimal), que puede utilizar como un contador en un bucle para obtener el resto de las respuestas. El recuento incluye este primer mensaje.
- El código de retorno del preprocesador de mandatos.
- Un código de razón, que es el código de razón del procesador de mandatos.

Este mensaje no contiene una CPF.

Por ejemplo:

```
CSQN205I    COUNT=    4, RETURN=0000000C, REASON=00000008
```

El campo COUNT tiene 8 bytes de longitud y está justificado por la derecha. Siempre comienza en la posición 18, es decir, inmediatamente después de COUNT=. El campo RETURN tiene 8 bytes de longitud en hexadecimal de caracteres y va inmediatamente después de RETURN= en la posición 35. El campo REASON tiene 8 bytes de longitud en hexadecimal de caracteres y va inmediatamente después de REASON= en la posición 52.

Si el valor de RETURN= es 00000000 y el valor de REASON= es 00000004, el conjunto de mensajes de respuesta está incompleto. Después de recuperar las respuestas indicadas por el mensaje CSQN205I, emita otra llamada MQGET para esperar un nuevo conjunto de respuestas. El primer mensaje del siguiente conjunto de respuestas es de nuevo CSQN205I, que indica cuántas respuestas hay, y si todavía quedan más por llegar.

Consulte la documentación [IBM MQ for z/OS mensajes, finalización, y códigos de razón](#) para obtener más información detallada sobre mensajes individuales.

Si está utilizando una característica de idioma no en inglés, el texto y el diseño de las respuestas son diferentes de los que se muestran aquí. Sin embargo, el tamaño y la posición del recuento y los códigos de retorno en el mensaje CSQN205I son los mismos.

## Si no recibe una respuesta

Hay una serie de pasos que puede seguir si no recibe una respuesta a la solicitud al servidor de mandatos.

Si no recibe una respuesta a su mensaje de solicitud, complete esta lista de comprobación:

- ¿Se está ejecutando el servidor de mandatos?
- ¿El valor de *WaitInterval* es lo suficiente largo?
- ¿Están definidas correctamente las colas de entrada de mandatos del sistema y de respuestas?
- ¿Fueron satisfactorias las llamadas MQOPEN a estas colas?
- ¿Está la cola de entrada de mandatos del sistema y la cola de respuesta habilitadas para llamadas MQPUT y MQGET?
- ¿Ha considerado la posibilidad de aumentar los atributos MAXDEPTH y MAXMSGL de las colas?
- ¿Está utilizando los campos *CorrelId* y *MsgId* correctamente?
- ¿El gestor de colas sigue en ejecución?
- ¿Se ha creado correctamente el mandato?
- ¿Están todos los enlaces remotos definidos y funcionando correctamente?

- ¿Estaban las llamadas MQPUT definidas correctamente?
- ¿Se ha definido la cola de respuesta como una cola dinámica temporal en lugar de una cola dinámica permanente? (Si el mensaje de solicitud es persistente, debe utilizar una cola dinámica permanente para la respuesta.)

Cuando el servidor de mandatos genera respuestas pero no puede grabarlas en la cola de respuesta que se especifica, las graba en la cola de mensajes no entregados.

## **z/OS** Pasar mandatos utilizando MGCRE

Con la autorización adecuada, un programa de aplicación puede realizar solicitudes a varios gestores de colas utilizando una rutina de servicio z/OS.

Si tiene la autorización correcta, puede pasar mandatos de IBM MQ desde el programa a varios gestores de colas mediante el servicio de z/OS MGCRE (SVC 34). El valor de la CPF identifica el gestor de colas concreto al que va dirigido el mandato. Para obtener información sobre las CPF, consulte [Identificadores de usuario para la seguridad de mandatos y de recursos de mandatos](#) y [“Ejecución de mandatos de gestor de colas en z/OS”](#) en la página 446.

Si utiliza MGCRE, puede utilizar una Señal de mandato y respuesta (CART) para obtener las respuestas directas para el mandato.

## **z/OS** Ejemplos de mandatos y sus respuestas

Utilice este tema como una serie de ejemplos de mandatos para el servidor de mandatos y las respuestas del servidor de mandatos.

Estos son algunos ejemplos de mandatos que se pueden incorporar en mensajes de IBM MQ, y los mensajes de usuario que son las respuestas. A menos que se indique lo contrario, cada línea de la respuesta es un mensaje independiente.

- [Mensajes de un mandato DEFINE](#)
- [Mensajes de un mandato DELETE](#)
- [Mensajes de mandatos DISPLAY](#)
- [Mensajes de mandatos con CMDSCOPE](#)
- [Mensajes de mandatos que generan mandatos con CMDSCOPE](#)

### **Mensajes de un mandato DEFINE**

El siguiente mandato:

```
DEFINE QLOCAL(Q1)
```

genera estos mensajes:

```
CSQN205I    COUNT=    2, RETURN=00000000, REASON=00000000
CSQ9022I +CSQ1 CSQMMSGP ' DEFINE QLOCAL' NORMAL COMPLETION
```

Estos mensajes de respuesta se producen en la terminación normal.

### **Mensajes de un mandato DELETE**

El siguiente mandato:

```
DELETE QLOCAL(Q2)
```

genera estos mensajes:

```
CSQN205I    COUNT=    4, RETURN=0000000C, REASON=00000008  
CSQM125I +CSQ1 CSQMUQLC QLOCAL (Q2) QSGDISP(QMGR) WAS NOT FOUND  
CSQM090E +CSQ1 CSQMUQLC FAILURE REASON CODE X'00D44002'  
CSQ9023E +CSQ1 CSQMUQLC ' DELETE QLOCAL' ABNORMAL COMPLETION
```

Estos mensajes indican que no existe una cola local llamada Q2.

### Mensajes de mandatos DISPLAY

Los ejemplos siguientes muestran las respuestas de algunos mandatos DISPLAY.

#### Averiguar el nombre de la cola de mensajes no entregados

Si desea averiguar el nombre de la cola de mensajes no entregados para un gestor de colas, emita este mandato desde un programa de aplicación:

```
DISPLAY QMGR DEADQ
```

Se devuelven los siguientes tres mensajes de usuario, de los que puede extraer el nombre necesario:

```
CSQN205I    COUNT=    3, RETURN=00000000, REASON=00000000  
CSQM409I +CSQ1 QMNAME(CSQ1) DEADQ(SYSTEM.DEAD.QUEUE  
CSQ9022I +CSQ1 CSQMDRTS ' DISPLAY QMGR' NORMAL COMPLETION
```

### Mensajes del mandato DISPLAY QUEUE

Los ejemplos siguientes muestran cómo el resultado de un mandato depende de los atributos especificados en el mandato.

#### Ejemplo 1

Defina una cola local mediante el mandato:

```
DEFINE QLOCAL(Q1) DESCR('A sample queue') GET(ENABLED) SHARE
```

Si emite el siguiente mandato desde un programa de aplicación:

```
DISPLAY QUEUE(Q1) SHARE GET DESCR
```

se devuelven estos tres mensajes de usuario:

```

CSQN205I   COUNT=    3, RETURN=00000000, REASON=00000000
CSQM401I +CSQ1 QUEUE(Q1                                ) TYPE(
QLOCAL ) QSGDISP(QMGR  )
DESCR(A sample queue
) SHARE GET(ENABLED )
CSQ9022I +CSQ1 CSQMMSG ' DISPLAY QUEUE' NORMAL COMPLETION

```

**Nota:** El segundo mensaje, CSQM401I, se muestra aquí ocupando cuatro líneas.

## Ejemplo 2

Dos colas tienen nombres que empiezan por la letra A:

- A1 es una cola local con su atributo PUT establecido en DISABLED.
- A2 es una cola remota con su atributo PUT establecido en ENABLED.

Si emite el siguiente mandato desde un programa de aplicación:

```

DISPLAY QUEUE(A*) PUT

```

se devuelven estos cuatro mensajes de usuario:

```

CSQN205I   COUNT=    4, RETURN=00000000, REASON=00000000
CSQM401I +CSQ1 QUEUE(A1                                ) TYPE(
QLOCAL ) QSGDISP(QMGR  )
PUT(DISABLED )
CSQM406I +CSQ1 QUEUE(A2                                ) TYPE(
QREMOTE ) PUT(ENABLED )
CSQ9022I +CSQ1 CSQMMSG ' DISPLAY QUEUE' NORMAL COMPLETION

```

**Nota:** El segundo y tercer mensaje, CSQM401I y CSQM406I, se muestran aquí ocupando tres y dos líneas.

## Mensajes del mandato DISPLAY NAMELIST

Defina una lista de nombres mediante el mandato:

```

DEFINE NAMELIST(N1) NAMES(Q1,SAMPLE_QUEUE)

```

Si emite el siguiente mandato desde un programa de aplicación:

```

DISPLAY NAMELIST(N1) NAMES NAMCOUNT

```

se devuelven los siguientes tres mensajes de usuario:

```

CSQN205I   COUNT=    3, RETURN=00000000, REASON=00000000
CSQM407I +CSQ1 NAMELIST(N1                               ) QS
GDISP(QMGR  ) NAMCOUNT(    2) NAMES(Q1
,SAMPLE_QUEUE
)
CSQ9022I +CSQ1 CSQMMSG ' DISPLAY NAMELIST' NORMAL COMPLETION

```

**Nota:** El segundo mensaje, CSQM407I, se muestra aquí ocupando tres líneas.

## Mensajes de mandatos con CMDSCOPE

Los ejemplos siguientes muestran las respuestas de mandatos que se han especificado con el atributo CMDSCOPE.

### Mensajes del mandato ALTER PROCESS

El siguiente mandato:

```
ALT PRO(V4) CMDSCOPE(*)
```

genera los siguientes mensajes:

```
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQN137I !MQ25 'ALT PRO' command accepted for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 5, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'ALT PRO' command responses from MQ26
CSQM125I !MQ26 CSQMMSGP PROCESS(V4) QSGDISP(QMGR) WAS NOT FOUND
CSQM090E !MQ26 CSQMMSGP FAILURE REASON CODE X'00D44002'
CSQ9023E !MQ26 CSQMMSGP 'ALT PRO' ABNORMAL COMPLETION
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'ALT PRO' command responses from MQ25
CSQ9022I !MQ25 CSQMMSGP 'ALT PRO' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=0000000C, REASON=00000008
CSQN123E !MQ25 'ALT PRO' command for CMDSCOPE(*) abnormal completion
```

Estos mensajes indican que el mandato se ha entrado en el gestor de colas MQ25 y se ha enviado a dos gestores de colas (MQ25 y MQ26). El mandato se ha ejecutado satisfactoriamente en MQ25, pero la definición de proceso no existía en MQ26, por lo que el mandato ha fallado en ese gestor de colas.

### Mensajes del mandato DISPLAY PROCESS

El siguiente mandato:

```
DIS PRO(V*) CMDSCOPE(*)
```

genera los siguientes mensajes:

```
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQN137I !MQ25 'DIS PRO' command accepted for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 5, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DIS PRO' command responses from MQ26
CSQM408I !MQ26 PROCESS(V2) QSGDISP(COPY)
CSQM408I !MQ26 PROCESS(V3) QSGDISP(QMGR)
CSQ9022I !MQ26 CSQMDRTS 'DIS PROCESS' NORMAL COMPLETION
CSQN205I COUNT= 7, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DIS PRO' command responses from MQ25
CSQM408I !MQ25 PROCESS(V2) QSGDISP(COPY)
CSQM408I !MQ25 PROCESS(V2) QSGDISP(GROUP)
CSQM408I !MQ25 PROCESS(V3) QSGDISP(QMGR)
CSQM408I !MQ25 PROCESS(V4) QSGDISP(QMGR)
CSQ9022I !MQ25 CSQMDRTS 'DIS PROCESS' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQN122I !MQ25 'DIS PRO' command for CMDSCOPE(*) normal completion
```

Estos mensajes indican que el mandato se ha entrado en el gestor de colas MQ25 y se ha enviado a dos gestores de colas (MQ25 y MQ26). Se muestra información sobre todos los procesos en cada gestor de colas que tienen nombres que empiezan por la letra V.

## Mensajes del mandato DISPLAY CHSTATUS

El siguiente mandato:

```
DIS CHS(VT) CMDSCOPE(*)
```

genera los siguientes mensajes:

```
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQN137I !MQ25 'DIS CHS' command accepted for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 4, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DIS CHS' command responses from MQ25
CSQM422I !MQ25 CHSTATUS(VT) CHLDISP(PRIVATE) CONNAME( ) CURRENT STATUS(STOPPED)
CSQ9022I !MQ25 CSQXDRTS ' DIS CHS' NORMAL COMPLETION
CSQN205I COUNT= 4, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DIS CHS' command responses from MQ26
CSQM422I !MQ26 CHSTATUS(VT) CHLDISP(PRIVATE) CONNAME( ) CURRENT STATUS(STOPPED)
CSQ9022I !MQ26 CSQXDRTS ' DIS CHS' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQN122I !MQ25 'DIS CHS' command for CMDSCOPE(*) normal completion
```

Estos mensajes indican que el mandato se ha entrado en el gestor de colas MQ25 y se ha enviado a dos gestores de colas (MQ25 y MQ26). Se muestra información sobre el estado de canal en cada gestor de colas.

## Mensajes del mandato STOP CHANNEL

El siguiente mandato:

```
STOP CHL(VT) CMDSCOPE(*)
```

genera estos mensajes:

```
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQN137I !MQ25 'STOP CHL' command accepted for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'STOP CHL' command responses from MQ25
CSQM134I !MQ25 CSQMTCHL STOP CHL(VT) COMMAND ACCEPTED
SQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'STOP CHL' command responses from MQ26
CSQM134I !MQ26 CSQMTCHL STOP CHL(VT) COMMAND ACCEPTED
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'STOP CHL' command responses from MQ26
CSQ9022I !MQ26 CSQXCRPS ' STOP CHL' NORMAL COMPLETION
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'STOP CHL' command responses from MQ25
CSQ9022I !MQ25 CSQXCRPS ' STOP CHL' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQN122I !MQ25 'STOP CHL' command for CMDSCOPE(*) normal completion
```

Estos mensajes indican que el mandato se ha entrado en el gestor de colas MQ25 y se ha enviado a dos gestores de colas (MQ25 y MQ26). El canal VT se ha detenido en cada gestor de colas.

## Mensajes de mandatos que generan mandatos con CMDSCOPE

El siguiente mandato:

```
DEF PRO(V2) QSGDISP(GROUP)
```

genera estos mensajes:

```
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQM122I !MQ25 CSQMMSGP 'DEF PRO' COMPLETED FOR QSGDISP(GROUP)
CSQN138I !MQ25 'DEFINE PRO' command generated for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DEFINE PRO' command responses from MQ25
CSQ9022I !MQ25 CSQMMSGP 'DEFINE PROCESS' NORMAL COMPLETION
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DEFINE PRO' command responses from MQ26
CSQ9022I !MQ26 CSQMMSGP 'DEFINE PROCESS' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQN122I !MQ25 'DEFINE PRO' command for CMDSCOPE(*) normal completion
```

Estos mensajes indican que el mandato se ha entrado en el gestor de colas MQ25. Cuando se creó el objeto en el repositorio compartido, se generó otro mandato y se envió a todos los gestores de colas activos del grupo de compartición de colas (MQ25 y MQ26).

z/OS

## Gestión de recursos de IBM MQ en z/OS

Utilice los enlaces de este tema para descubrir cómo gestionar los recursos utilizados por IBM MQ for z/OS, por ejemplo, gestionar archivos de registro, conjuntos de datos, conjuntos de páginas, agrupaciones de almacenamiento intermedio y emparejamiento de estructuras de recurso.

Utilice los siguientes enlaces para obtener información detallada de las distintas tareas administrativas que podría tener que completar al utilizar IBM MQ for z/OS:

- [“Gestión de registros” en la página 480](#)
- [“Gestión del conjunto de datos de rutina de carga \(BSDS\)” en la página 489](#)
- [“Gestión de conjuntos de páginas” en la página 497](#)
- [“Cómo hacer una copia de seguridad y recuperar conjuntos de páginas” en la página 504](#)
- [“Cómo hacer una copia de seguridad y restaurar las colas utilizando CSQUTIL” en la página 508](#)
- [“Gestión de agrupaciones de almacenamientos intermedios” en la página 509](#)
- [“Gestión de grupos de compartición de colas y colas compartidas en z/OS” en la página 510](#)

### Conceptos relacionados

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

[“Administración de IBM MQ for z/OS” en la página 434](#)

La administración de gestores de colas y recursos asociados incluye las tareas que se realizan con frecuencia para activar y gestionar estos recursos. Elija el método que prefiera para administrar los gestores de colas y los recursos asociados.

[“Emisión de mandatos a IBM MQ for z/OS” en la página 434](#)

Puede utilizar mandatos de script de IBM MQ (MQSC) en modalidad interactiva o por lotes para controlar un gestor de colas.

[“Recuperación y reinicio en z/OS” en la página 520](#)

En este tema se explican los mecanismos de recuperación y reinicio utilizados por IBM MQ.

### Tareas relacionadas

[Planificación del entorno de IBM MQ en z/OS](#)

[Configuración de gestores de colas en z/OS](#)

[Utilización de los programas de utilidad de IBM MQ for z/OS](#)

## Referencia relacionada

“Orígenes desde los que puede emitir mandatos MQSC en z/OS” en la página 20  
Los mandatos MQSC se pueden emitir desde varios orígenes, en función del mandato.

“Los programas de utilidad de IBM MQ for z/OS” en la página 443

IBM MQ for z/OS proporciona un conjunto de programas de utilidad que puede utilizar para ayudar en la administración del sistema.

Referencia de formatos de mandato programable

## z/OS Gestión de registros

Utilice este tema para aprender a gestionar los archivos de registro de IBM MQ, incluyendo el proceso de archivado de registros, la utilización de la compresión de registros de anotaciones, la recuperación de registros de anotaciones y la impresión de registros de anotaciones.

En este tema se describen las tareas relacionadas con la gestión de los registros de IBM MQ. Contiene las siguientes secciones:

## z/OS Archivado de registros con el mandato ARCHIVE LOG

Un operador autorizado puede archivar los conjuntos de datos de registro activo actual de IBM MQ siempre que sea necesario utilizando el mandato **ARCHIVE LOG**.

Cuando se emite el mandato **ARCHIVE LOG**, IBM MQ trunca los conjuntos de datos de registro activo actual, luego ejecuta un proceso de descarga asíncrono y actualiza el BSDS con un registro del proceso de descarga.

El mandato **ARCHIVE LOG** tiene una opción **MODE (QUIESCE)**. Con esta opción, los trabajos y los usuarios de IBM MQ se desactivan temporalmente después de un punto de confirmación, y el punto de coherencia resultante se captura en el registro activo actual antes de descargarlo.

Considere la posibilidad de utilizar la opción **MODE (QUIESCE)** cuando planifique una estrategia de copia de seguridad para la recuperación fuera del sitio. Esta opción crea un punto de coherencia para todo el sistema, lo que minimiza el número de incoherencias en los datos cuando el registro de archivado se utiliza con la copia del conjunto de páginas de copia de seguridad más actual durante la recuperación. Por ejemplo:

```
ARCHIVE LOG MODE(QUIESCE)
```

Si emite el mandato **ARCHIVE LOG** sin especificar un parámetro **TIME**, el periodo de tiempo de desactivación temporal toma de forma predeterminada el valor del parámetro **QUIESCE** de la macro CSQ6ARVP. Si el tiempo necesario para que se complete el mandato **ARCHIVE LOG MODE (QUIESCE)** es inferior al tiempo especificado, el mandato se ejecuta satisfactoriamente; de lo contrario, el mandato falla cuando finaliza el periodo de tiempo. Puede especificar el período de tiempo explícitamente utilizando la opción **TIME**, por ejemplo:

```
ARCHIVE LOG MODE(QUIESCE) TIME(60)
```

Este mandato especifica un período de desactivación temporal de hasta 60 segundos antes de que se produzca el proceso de **ARCHIVE LOG**.

**Atención:** El uso de la opción **TIME** cuando el tiempo es crucial puede interrumpir considerablemente la disponibilidad de IBM MQ para todos los trabajos y usuarios que utilizan recursos de IBM MQ.

De forma predeterminada, el mandato se procesa de forma asíncrona desde el momento en que somete el mandato. (Para procesar el mandato de forma síncrona con otros mandatos de IBM MQ, utilice la opción **WAIT (YES)** con **QUIESCE**, pero tenga en cuenta que la consola de z/OS está bloqueada para la entrada de mandatos de IBM MQ durante todo el periodo **QUIESCE**).

Durante el período de desactivación temporal:



- Los trabajos y los usuarios en el gestor de colas están autorizados a realizar el proceso de confirmación, pero serán suspendidos si intentan actualizar cualquier recurso de IBM MQ después de la confirmación.
- Los trabajos y los usuarios que sólo leen datos pueden verse afectados, ya que podrían estar a la espera de bloqueos mantenidos por trabajos o usuarios que fueron suspendidos.
- Se pueden iniciar nuevas tareas, pero estas no pueden actualizar datos.

La salida del mandato **DISPLAY LOG** utiliza el mensaje CSQV400I para indicar que hay una inmovilización en curso.

**V 9.1.4** Por ejemplo, a partir de IBM MQ 9.1.4:

```
CSQJ322I +CSQ1 DISPLAY LOG report ...
Parameter  Initial value      SET value
-----
INBUFF     60
OUTBUFF    400
MAXRTU     2
MAXARCH    2
TWOACTV    YES
TWOARCH    YES
TWOBSDS    YES
OFFLOAD    YES
MAXCNOFF   0
WRTHRSR   20
DEALLCT    0
COMPLOG    NONE
ZHYWRITE   NO
End of LOG report
CSQJ370I +CSQ1 LOG status report ...
Copy %Full zHyperWrite Encrypted DSName
  1   68 NO          NO          VICY.CSQ1.LOGCOPY1.DS01
  2   68 NO          NO          VICY.CSQ1.LOGCOPY2.DS01
Restarted at 2019-08-15 09:49:30 using RBA=000000000891B000
Latest RBA=000000000891CCF8
Offload task is AVAILABLE
Full logs to offload - 0 of 4
CSQV400I +CSQ1 ARCHIVE LOG QUIESCE CURRENTLY ACTIVE
CSQ9022I +CSQ1 CSQJC001 ' DISPLAY LOG' NORMAL COMPLETION
```

Por ejemplo, para Long Term Support y Continuous Delivery anterior a IBM MQ 9.1.2:

```
CSQJ322I +CSQ1 DISPLAY LOG report ...
Parameter  Initial value      SET value
-----
INBUFF     60
OUTBUFF    400
MAXRTU     2
MAXARCH    2
TWOACTV    YES
TWOARCH    YES
TWOBSDS    YES
OFFLOAD    YES
MAXCNOFF   0
WRTHRSR   20
DEALLCT    0
COMPLOG    NONE
ZHYWRITE   NO                      YES
End of LOG report
CSQJ370I +CSQ1 LOG status report ...
Copy %Full PPRC DSName
  1   68 NO          VICY.CSQ1.LOGCOPY1.DS01
  2   68 NO          VICY.CSQ1.LOGCOPY2.DS01
Restarted at 2014-04-15 09:49:30 using RBA=000000000891B000
Latest RBA=000000000891CCF8
Offload task is AVAILABLE
Full logs to offload - 0 of 4
CSQV400I +CSQ1 ARCHIVE LOG QUIESCE CURRENTLY ACTIVE
CSQ9022I +CSQ1 CSQJC001 ' DISPLAY LOG' NORMAL COMPLETION
```

Cuando todas las actualizaciones se han desactivado temporalmente, el registro de historial de desactivación temporal en el BSDS se actualiza con la fecha y hora en que se truncaron los conjuntos de datos de registro activo, y con la última RBA grabada en los conjuntos de datos de registro activo

actual. IBM MQ trunca los conjuntos de datos de registro activo actual, cambia a los siguientes conjuntos de datos de registro activo disponibles, y emite el mensaje CSQJ311I que indica que el proceso de descarga se ha iniciado.

Si las actualizaciones no se pueden desactivar temporalmente antes de que caduque el periodo de inmovilización, IBM MQ emite el mensaje CSQJ317I y el proceso de **ARCHIVE LOG** termina. Los conjuntos de datos de registro activo actual no se truncan, ni se cambia a los siguientes conjuntos de datos de registro disponibles, y el proceso de descarga no se inicia.

Tanto si la desactivación temporal ha sido satisfactoria como si no, todos los usuarios y trabajos suspendidos se reanudan, y IBM MQ emite el mensaje CSQJ312I, que indica que se ha finalizado la desactivación temporal y se ha reanudado la actividad de actualización.

Si **ARCHIVE LOG** se emite cuando el registro activo actual es el último conjunto de datos de registro activo disponible, el mandato no se procesa y IBM MQ emite el siguiente mensaje:

```
CSQJ319I - csect-name CURRENT ACTIVE LOG DATA SET IS THE LAST
AVAILABLE ACTIVE LOG DATA SET. ARCHIVE LOG PROCESSING
WILL BE TERMINATED
```

Si **ARCHIVE LOG** se emite cuando ya hay otro mandato **ARCHIVE LOG** en curso, el nuevo mandato no se procesa, y IBM MQ emite el siguiente mensaje:

```
CSQJ318I - ARCHIVE LOG COMMAND ALREADY IN PROGRESS
```

Para obtener información sobre los mensajes que se emiten durante el archivado, consulte [Mensajes para IBM MQ for z/OS](#).

## Reiniciar el proceso de archivado de registro después de una anomalía

Si hay un problema durante el proceso de archivado de registro (por ejemplo, un problema con la asignación o los montajes de cinta), el archivado del registro activo puede suspenderse. Puede cancelar el proceso de archivado y reiniciarlo utilizando el mandato siguiente:

```
ARCHIVE LOG CANCEL OFFLOAD
```

Este mandato cancela cualquier proceso de descarga actualmente en curso y reinicia el proceso de archivado. Empieza con el conjunto de datos de registro más antiguo que no se ha archivado, y prosigue con todos los conjuntos de datos de registro activo que necesitan descarga. Todas las operaciones de archivado de registro que se han suspendido se reinician.

Utilice este mandato sólo si está seguro de que la tarea de archivado de registro actual ya no funciona, o si desea reiniciar un intento anterior que ha fallado. Esto se debe a que el mandato puede provocar una terminación anómala de la tarea de descarga, lo que podría dar lugar a un volcado.

### **Controlar el archivado y el registro**

Puede controlar la compresión, impresión, archivado, recuperación y registro cronológico utilizando las macros CSQ6LOGP, CSQ6ARVP y CSQ6SYSP. Tenga en cuenta que los cambios de objetos privados sólo se registran en los registros de IBM MQ. Los cambios en objetos GROUP (como los canales de entrada compartidos) también se registran, porque las definiciones se propagan por todo el grupo y se mantienen localmente.

Muchos aspectos del archivado y el registro cronológico se controlan mediante parámetros que se establecen con las macros CSQ6LOGP, CSQ6ARVP y CSQ6SYSP del módulo de parámetros del sistema cuando se personaliza el gestor de colas. Consulte [Ajustar el módulo de parámetros del sistema](#) para obtener detalles de estas macros.

Algunos de estos parámetros se pueden cambiar mientras se ejecuta un gestor de colas utilizando los mandatos IBM MQ MQSC SET LOG, SET SYSTEM y SET ARCHIVE. Se muestran en la [Tabla 33 en la página 483](#):

Tabla 33. Parámetros de archivado y registro cronológico que se pueden cambiar mientras un gestor de colas está en ejecución

Mandato SET	Parámetros
LOG	WRTHRSH, MAXARCH, DEALLCT, MAXRTU, COMPLOG
ARCHIVE	Todo
SYSTEM	LOGLOAD

Puede visualizar los valores de todos los parámetros mediante los mandatos MQSC [DISPLAY LOG](#), [DISPLAY ARCHIVE](#) y [DISPLAY SYSTEM](#). Estos mandatos también muestran información de estado sobre el archivado y el registro cronológico.

## Controlar la compresión de registros

Puede habilitar e inhabilitar la compresión de registros de anotaciones

- Utilizando los mandatos SET y DISPLAY LOG en MQSC; consulte [Mandatos MQSC](#)
- Invocando la interfaz PCF. Consulte [“Introducción a los Formatos de mandato programable de IBM MQ” en la página 25](#)
- Utilizando la macro CSQ6LOGP del módulo de parámetros del sistema; consulte [Utilización de CSQ6LOGP](#)

## Imprimir registros de anotaciones

Puede extraer e imprimir registros de anotaciones utilizando el programa de utilidad CSQ1LOGP. Para obtener instrucciones, consulte [El programa de utilidad de impresión de registro](#).

## Recuperar registros

Normalmente no es necesario hacer copias de seguridad y restaurar los registros de IBM MQ, especialmente si está utilizando el registro dual. Sin embargo, en circunstancias excepcionales, como un error de E/S en un registro, es posible que necesite recuperar los registros. Utilice los Servicios de método de acceso para suprimir y volver a definir el conjunto de datos y luego copie el registro dual correspondiente en el mismo.

### Descartar conjuntos de datos de registro de archivado

Puede descartar los conjuntos de datos de registro de archivado y elegir si desea descartar los registros de forma automática o manual.

Debe mantener suficientes datos de registro para poder realizar la recuperación de unidad de trabajo, la recuperación desde soporte de almacenamiento de conjunto de páginas si se pierde un conjunto de páginas o la recuperación desde soporte de almacenamiento de estructura CF si se pierde una estructura CF. No descarte conjuntos de datos de registro de archivado que puedan ser necesarios para la recuperación; si descarta estos conjuntos de datos de registro de archivado, es posible que no pueda realizar las operaciones de recuperación necesarias.

Si ha confirmado que los conjuntos de datos de registro de archivado se pueden descartar, puede hacerlo de una de las siguientes maneras:

- [Supresión automática de conjuntos de datos de registro de archivado](#)
- [Supresión manual de conjuntos de datos de registro de archivado](#)

## Supresión automática de conjuntos de datos de registro de archivado

Puede utilizar un DASD o un sistema de gestión de cintas para suprimir conjuntos de datos de registro de archivado automáticamente. El período de retención para los conjuntos de datos de registro de archivado de IBM MQ se especifica mediante el campo de período de retención ARCRETN de la macro de instalación CSQ6ARVP (consulte [Utilización de CSQ6ARVP](#) para obtener más información).

El valor predeterminado para el periodo de retención especifica que los registros de archivado se deben mantener durante 9999 días (el máximo posible).

**Importante:** Puede cambiar el periodo de retención, pero debe asegurarse de que puede acomodar el número de ciclos de copia de seguridad que ha previsto.

IBM MQ utiliza el valor del periodo de retención como el valor para el parámetro de JCL RETPD cuando se crean conjuntos de datos de registro de archivado.

El periodo de retención establecido por el subsistema de gestión de almacenamiento (SMS) de MVS/DFP se puede alterar temporalmente con este parámetro de IBM MQ. Normalmente, el período de retención se establece en el valor más pequeño especificado por IBM MQ o SMS. El administrador de almacenamiento y el administrador de IBM MQ deben acordar un valor de período de retención que sea adecuado para IBM MQ.

**Nota:** IBM MQ no tiene un método automatizado para suprimir información sobre conjuntos de datos de registro de archivado del BSDS, ya que algunos sistemas de gestión de cintas proporcionan alteraciones manuales externas de períodos de retención. Por lo tanto, puede que todavía haya información sobre un conjunto de datos de registro de archivado en el BSDS mucho tiempo después de que haya finalizado el periodo de retención de conjunto de datos y de que el sistema de gestión de cintas haya borrado el conjunto de datos. Y a la inversa, es posible que se haya excedido el número máximo de conjuntos de datos de registro de archivado y los datos del BSDS se pueden haber descartado antes de que el conjunto de datos haya alcanzado su fecha de caducidad.

Si los conjuntos de datos de registro de archivado se suprimen automáticamente, recuerde que la operación no actualiza la lista de registros de archivado en el BSDS. Puede actualizar el BSDS con el programa de utilidad de inventario de registro de cambios, tal como se describe en [“Cambiar el BSDS”](#) en la [página 491](#). La actualización no es esencial. Registrar registros de archivado antiguos desaprovecha espacio en el BSDS, pero no causa otros daños.

## Supresión manual de conjuntos de datos de registro de archivado

Debe mantener todos los registros de anotaciones que se remontan a la RBA más baja identificada en los mensajes CSQI024I y CSQI025I. Esta RBA se obtiene mediante el mandato DISPLAY USAGE que emitió al crear un punto de recuperación utilizando el [Método 1: Copia se seguridad completa](#).

**Lea el tema [Creación de un punto de recuperación para recursos no compartidos antes de descartar cualquier registro](#).**

### Localizar y descartar conjuntos de datos de registro de archivado

Después de haber establecido la RBA de registro mínima necesaria para la recuperación, puede buscar conjuntos de datos de registro de archivado que contengan sólo registros de anotaciones anteriores realizando el siguiente procedimiento:

1. Utilice el programa de utilidad imprimir correlación de registros para imprimir el contenido del BSDS. Para ver un ejemplo de la salida, consulte [El programa de utilidad de correlación de registro de impresión](#).
2. Busque las secciones de la salida llamada ARCHIVE LOG COPY n DATA SETS. Si utiliza el registro dual, hay dos secciones. Las columnas etiquetadas STARTRBA y ENDRBA muestran el rango de las RBA contenidas en cada volumen. Busque los volúmenes con rangos que incluyan la RBA mínima que ha encontrado con los mensajes CSQI024I y CSQI025I. Estos son los volúmenes más antiguos que tiene que mantener. Si está utilizando el registro dual, hay dos de esos volúmenes.

Si no hay volúmenes que tengan un rango apropiado, se aplica uno de los casos siguientes:

- La RBA mínima aún no se ha archivado, y puede descartar todos los volúmenes de registro de archivado.
- La lista de volúmenes de registro de archivado en el BSDS se reinició cuando el número de volúmenes excedió el número permitido por el parámetro MAXARCH de la macro CSQ6LOGP. Si el BSDS no registra un volumen de registro de archivado, ese volumen no se puede utilizar para la recuperación. Por lo tanto, considere la posibilidad de añadir información sobre los volúmenes existentes al BSDS. Si desea más instrucciones, consulte [“Cambios para registros de archivado” en la página 493](#).

Asimismo, considere la posibilidad de aumentar el valor de MAXARCH. Para obtener información, consulte [Utilización de CSQ6LOGP](#).

3. Suprima cualquier volumen o conjunto de datos de registro de archivado con un valor ENDRBA que sea inferior al valor STARTRBA del volumen más antiguo que desea mantener. Si está utilizando el registro dual, suprima las dos copias.

Puesto que las entradas del BSDS se reinician, las primeras entradas en la sección de registro de archivado del BSDS podrían ser más recientes que las entradas del final. Mire la combinación de fecha y hora y compare su antigüedad. No presuponga que puede descartar todas las entradas anteriores a la entrada del registro de archivado que contiene la LOGRBA mínima.

Suprima los conjuntos de datos. Si los archivos están en cinta, borre las cintas. Si están en DASD, ejecute un programa de utilidad de z/OS para suprimir cada conjunto de datos. A continuación, si desea que el BSDS liste únicamente volúmenes de archivado existentes, utilice el programa de utilidad de inventario de registro de cambios (CSQJU003) para suprimir entradas para los volúmenes descartados. Consulte [“Cambios para registros de archivado” en la página 493](#) para obtener un ejemplo.

### **El efecto de la derivación de registro**

Las transacciones de larga ejecución pueden generar registros de anotaciones de unidad de trabajo que abarcan conjuntos de datos de registro. IBM MQ maneja este escenario utilizando la derivación de registros, una técnica que mueve los registros de anotaciones para optimizar la cantidad de datos de registro retenidos y el tiempo de reinicio del gestor de colas.

Cuando una unidad de trabajo se considera que es larga, se graba una representación de cada registro de anotaciones más abajo en el registro. Esto se conoce como *derivación de registro*. Se describe de forma más completa en [Archivos de registro](#).

El gestor de colas utiliza estos registros de anotaciones derivados en lugar de los originales después de una anomalía, para asegurar la integridad de la unidad de trabajo. Esto presenta dos ventajas:

- se reduce la cantidad de datos de registro que deben conservarse para la coordinación de unidades de trabajo
- se deben recorrer menos datos de registro durante el reinicio del gestor de colas, por lo que el gestor de colas se reinicia más rápidamente

Los registros de anotaciones derivados no contienen información suficiente para las operaciones de recuperación desde soporte de almacenamiento.

Los datos contenidos en el registro se utilizan para dos fines diferentes; la recuperación desde soporte de almacenamiento y la coordinación de unidades de trabajo. Si se produce una anomalía de soporte que afecta a una estructura CF o a un conjunto de páginas, el gestor de colas puede recuperar el soporte en el punto de la anomalía restaurando una copia anterior y actualizando ésta utilizando datos contenidos en el registro. La actividad persistente realizada en una unidad de trabajo se registra en el registro para que, en el caso de una anomalía, pueda ser restituida o se puedan recuperar los bloqueos en los recursos cambiados. La cantidad de datos de registro que necesita conservar para permitir la recuperación del gestor de colas se ve afectada por estos dos elementos.

Para la recuperación desde soporte de almacenamiento, debe conservar datos de registro suficientes para poder realizar la recuperación desde soporte de almacenamiento desde al menos la copia de

soporte de almacenamiento más reciente y para poder restituir. (Su sitio puede estipular la capacidad de recuperar desde copias de seguridad más antiguas.) Para la integridad de la unidad de trabajo, debe conservar los datos de registro para sus unidades de trabajo en curso o pendientes más antiguas.

Para ayudarle en la gestión del sistema, el gestor de colas detecta unidades de trabajo antiguas en cada archivo de registro e informa sobre ellas en los mensajes CSQJ160 y CSQJ161. Una tarea interna lee información de registro de unidad de trabajo para estas unidades de trabajo antiguas y la reescribe de forma más sucinta en la posición actual del registro. El mensaje CSQR026 indica cuándo ha sucedido esto. El mandato MQSC DISPLAY USAGE TYPE(DATASET) también puede ayudarle a gestionar la retención de datos de registro. El mandato informa sobre tres fragmentos de información de recuperación:

1. La cantidad del registro que debe conservarse para la recuperación de la unidad de trabajo.
2. La cantidad del registro que debe conservarse para la recuperación desde soporte de almacenamiento de conjuntos de páginas.
3. Para un gestor de colas en un grupo de compartición de colas, la cantidad del registro que debe conservarse para la recuperación desde soporte de almacenamiento de estructuras CF.

Para cada uno de estos fragmentos de información, se intenta correlacionar los datos de registro más antiguos necesarios con un conjunto de datos. A medida que se inician y detienen nuevas unidades de trabajo, cabría esperar que el punto (1) se mueva a una posición más reciente en el registro. Si no se mueve, los mensajes de unidad de trabajo de larga ejecución le avisan de que hay un problema. (2) se refiere a la recuperación desde el soporte de almacenamiento de conjuntos de páginas si el gestor de colas se concluyera ahora y se reiniciara. No tiene ninguna información sobre cuándo realizó la última copia de seguridad de los conjuntos de páginas, o qué copia de seguridad tendría que utilizar si hubiese una anomalía de conjunto de páginas. Normalmente se mueve a una posición más reciente en el registro durante el proceso de punto de comprobación cuando los cambios contenidos en las agrupaciones de almacenamiento intermedio se graban en los conjuntos de páginas. En (3), el gestor de colas tiene información sobre las copias de seguridad de estructura CF realizadas en este gestor de colas o en otros gestores de colas del grupo de compartición de colas. Sin embargo, la recuperación de la estructura CF requiere una fusión de datos de registro de todos los gestores de colas del grupo de compartición de colas que han interactuado con la estructura CF desde la última copia de seguridad. Esto significa que los datos de registro se identifican mediante un número de secuencia de registro de anotaciones (o LRSN), que está basado en la indicación de fecha y hora y que, por lo tanto, es aplicable en todo el grupo de compartición de colas, en lugar de una RBA que sería diferente en diferentes gestores de colas del grupo de compartición de colas. Normalmente se mueve a una posición más reciente en el registro cuando se ejecutan mandatos BACKUP CFSTRUCT en este o en otros gestores de colas del grupo de compartición de colas.

## **Restablecer el registro del gestor de colas**

Utilice este tema para comprender cómo restablecer el registro del gestor de colas.

No debe permitir que la RBA de registro del gestor de colas se ajuste desde el final de rango de RBA de registro a 0, ya que esto creará una interrupción del gestor de colas y no se podrán recuperar todos los datos persistentes. El final de la RBA de registro es un valor de FFFFFFFFFF (si se utilizan las RBA de 6 bytes) o FFFFFFFFFFFFFFFF (si se utilizan las RBA de 8 bytes).

El gestor de colas emite los mensajes [CSQI045I](#), [CSQI046E](#), [CSQI047E](#), [CSQJ031D](#) y [CSQJ032E](#) para indicar que el rango de registro utilizado es importante y que se debe planificar una medida para evitar una interrupción no planificada.

El gestor de colas finaliza con el código de razón [00D10257](#) cuando el valor de RBA alcanza FFF800000000 (si se utilizan las RBA de 6 bytes) o FFFFFFFC00000000 (si se utilizan las RBA de 8 bytes).

Si se están utilizando las RBA de registro de 6 bytes, piense en convertir el gestor de colas para utilizar las RBA de registro de 8 bytes en lugar de restablecer el registro del gestor de colas, siguiendo el proceso descrito en [“Implementación de la RBA \(Relative Byte Address\)”](#) en la [página 487](#). Convertir un gestor de colas para que utilice las RBA de registro de 8 bytes requiere una interrupción más corta que si se restablece el registro y aumenta de forma importante el periodo de tiempo antes de que sea necesario restablecer el registro.



El mensaje CSQJ034I, emitido durante la inicialización del gestor de colas, indica el fin del rango de RBA de registro para el gestor de colas, según su configuración, y se puede utilizar para determinar si se están utilizando las RBA de registro de 6 bytes o de 8 bytes.

Siga este procedimiento para restablecer el registro del gestor de colas:

1. Resuelva las unidades de trabajo no resueltas. El número de unidades de trabajo no resueltas se muestra en el inicio del gestor de colas en el mensaje CSQR005I como número de INDOUBT. En cada punto de comprobación y durante la conclusión del gestor de colas, el gestor de colas emite automáticamente el mandato

**DISPLAY CONN(\*) TYPE(CONN) ALL WHERE(UOWSTATE EQ UNRESOLVED)** para proporcionar información sobre las unidades de trabajo sin resolver.

Consulte [Cómo se resuelven las unidades de recuperación dudosas](#) para obtener información sobre cómo resolver unidades de recuperación. El último recurso es utilizar el mandato **RESOLVE INDOUBT** de MQSC para resolver manualmente unidades de recuperación pendientes.

2. Concluya ordenadamente el gestor de colas.

Puede utilizar **STOP QMGR** o **STOP QMGR MODE(FORCE)** ya que estos mandatos desechan las páginas modificadas de las agrupaciones de almacenamiento intermedio a los conjuntos de páginas.

3. Si un gestor de colas forma parte de un grupo de compartición de colas, realice copias de seguridad de CFSTRUCT en otros gestores de colas para todas las estructuras en el grupo de compartición de colas. De este modo, se garantiza que las copias de seguridad más recientes no estén en este registro del gestor de colas, y que este registro del gestor de colas no es necesario para la recuperación CFSTRUCT.
4. Defina nuevos registros y BSDS mediante CSQJU003 (consulte [El programa de utilidad de inventario de registro de cambios](#) para obtener más información sobre el uso del programa de utilidad de inventario de registro de cambios).
5. Ejecute **CSQUTIL RESETPAGE** para todos los conjuntos de páginas de este gestor de colas (consulte [Copia de una página y restablecimiento del registro](#) para obtener más información sobre esta función). Tenga en cuenta que las RBA de los conjuntos de páginas pueden restablecerse de forma independiente, de modo que puedan enviarse varios trabajos simultáneos (por ejemplo, uno por conjunto de páginas) para reducir el tiempo transcurrido de este paso.
6. Reinicie el gestor de colas

### Conceptos relacionados


[“Implementación de la RBA \(Relative Byte Address\)” en la página 487](#)

Antes de IBM MQ 8.0, IBM MQ for z/OS utilizaba una RBA de registro de 6 bytes para identificar la ubicación de datos dentro del registro. A partir de IBM MQ 8.0, la RBA de registro puede tener una longitud de 8 bytes, aumentando el periodo de tiempo antes de tener que restablecer el registro.

### **Implementación de la RBA (Relative Byte Address)**

Antes de IBM MQ 8.0, IBM MQ for z/OS utilizaba una RBA de registro de 6 bytes para identificar la ubicación de datos dentro del registro. A partir de IBM MQ 8.0, la RBA de registro puede tener una longitud de 8 bytes, aumentando el periodo de tiempo antes de tener que restablecer el registro.

Esta nueva característica se debe habilitar de forma explícita. Consulte en la sección [Planificación del aumento del rango máximo de registros direccionables](#) las consideraciones a tener en cuenta a la hora de planificar la habilitación de la RBA de registro de 8 bytes.

**Importante:**  Si el gestor de colas no forma parte de un grupo de compartición de colas y posteriormente vuelve a migrar a IBM MQ 8.0.0 o 9.0.0, habiendo habilitado la RBA de registro de 8 bytes, asegúrese de que está utilizando **OPMODE=NEWFUNC** en esos releases; de lo contrario, el gestor de colas no se iniciará.

Siga estas instrucciones, en el orden en que se muestran para habilitar la RBA de registro de 8 bytes en un gestor de colas de IBM MQ for z/OS individual:

1. **V 9.1.0** Si el gestor de colas está en un grupo de compartición de colas, asegúrese de que todos los gestores de colas del grupo de compartición de colas que no están en IBM MQ 9.1.0 se están ejecutando con **OPMODE=(NEWFUNC,800)**, o **OPMODE=(NEWFUNC,900)** antes de pasar al paso “2” en la [página 488](#).

No es necesario que realice una parada completa del grupo de compartición de colas para hacerlo. Puede detener cada gestor de colas que no esté en IBM MQ 9.1.0 a su vez, cambiarlo por **OPMODE=(NEWFUNC,800)** o **OPMODE=(NEWFUNC,900)** y reiniciarlo.

Una vez que todos los gestores de colas del grupo de compartición de colas se están ejecutando en IBM MQ 9.1.0, o con **OPMODE=(NEWFUNC,800)**, o **OPMODE=(NEWFUNC,900)**, realice los pasos siguientes para cada gestor de colas del grupo de compartición de colas, hasta que todos los gestores de colas se ejecutan con el nuevo BSDS.

2. Asigne nuevos conjuntos de datos BSDS con atributos similares al BSDS actual. Puede personalizar el ejemplo CSQ4BSDS y suprimir cualquier sentencia irrelevante o puede utilizar el JCL existente, pero cambiar el nombre de BSDS por algo parecido a ++HLQ++ .NEW .BSDS01.

#### Notas:

- a. Compruebe los atributos del nuevo BSDS antes de enviar el trabajo para asignar el nuevo BSDS. El único atributo que puede cambiar es el tamaño del BSDS.
  - b. El nuevo BSDS contiene más datos que el BSDS actual, por lo tanto, debe asegurarse de que los nuevos conjuntos de datos tengan asignado un espacio disponible suficiente. El JCL de ejemplo en thlqual.SCSQPROC(CSQ4BSDS) contiene los valores recomendados cuando se define un nuevo BSDS.
3. Concluya ordenadamente el gestor de colas.
  4. Ejecute el [Programa de utilidad de conversión de BSDS \(CSQJUCNV\)](#) para convertir los BSDS existentes en los nuevos conjuntos de datos BSDS. Normalmente esto tarda unos segundos en ejecutarse.

El BSDS existente no se modificará durante este proceso, y puede utilizarlo para la inicialización del gestor de colas en el caso de que la conversión no se realice satisfactoriamente.

5. Cambie el nombre del BSDS actual para que se convierta en el BSDS antiguo, y el nuevo BSDS para que se convierta en el BSDS actual, de modo que se utilicen los nuevos conjuntos de datos la próxima vez que reinicie el gestor de colas. Puede utilizar el mandato ALTER de AMS (Access Method Services) de DFSMS, por ejemplo:

```
ALTER '++HLQ++.BSDS01' NEWNAME('++HLQ++.OLD.BSDS01')
ALTER '++HLQ++.NEW.BSDS01' NEWNAME('++HLQ++.BSDS01')
```

Asegúrese de que también emite los mandatos para cambiar el nombre de los datos y de las partes del índice del clúster VSAM.

6. Reinicie el gestor de colas. Debe comenzar con el mismo periodo de tiempo como si utilizara la RBA de registro de 6 bytes.

Si el gestor de colas no se reinicia correctamente debido a un error de acceso al BSDS convertido, intente identificar la causa del error, resuelva el problema y vuelva a intentar la operación. Si es necesario, póngase con su centro de soporte de IBM para obtener ayuda.

Si es necesario, se puede restituir el cambio en este punto, realizando lo siguiente:

- a. Cambie el nombre del BSDS actual para convertirlo en el BSDS nuevo.
- b. Cambie el nombre del BSDS antiguo para convertirlo en el BSDS actual.
- c. Reinicie el gestor de colas.

Cuando el gestor de colas se haya reiniciado correctamente con el BSDS convertido, no intente iniciar el gestor de colas utilizando el BSDS.

7. Se emite el mensaje [CSQJ034I](#) durante la inicialización del gestor de colas para indicar el final de la RBA de registro para el gestor de colas como configurado. Confirme que el final del rango de la RBA



de registro visualizado sea FFFFFFFFFFFFFFFF. Esto indica que la RBA de registro de 8 bytes se está utilizando.

**Nota:** **V9.1.0** Para poder habilitar una RBA de registro de 8 bytes en un nuevo gestor de colas, antes de que se inicie por primera vez, primero debe crear una versión vacía 1 de formato BSDS y utilizarla como entrada al programa de utilidad de conversión BSDS para generar una versión 2 de formato BSDS. Consulte [Crear los conjuntos de datos de rutina de carga y de registro cronológico](#) para obtener información sobre cómo llevar a cabo este proceso.

### Conceptos relacionados

[RBA \(Dirección de byte relativa\) de registro mayor](#)

### Tareas relacionadas

[Planificación del aumento del rango máximo de registros direccionables](#)

### Referencia relacionada

[El programa de utilidad de conversión BSDS \(CSQJUCNV\)](#)

## **z/OS** **Gestión del conjunto de datos de rutina de carga (BSDS)**

El conjunto de datos de rutina de carga se utiliza para hacer referencia a conjuntos de datos de registro y registros de anotaciones. Lea este tema para saber cómo puede examinar, cambiar y recuperar el BSDS.

Para obtener más información, consulte [El conjunto de datos de rutina de carga](#).

En este tema se describen las tareas relacionadas con la gestión del conjunto de datos de rutina de carga. Contiene las siguientes secciones:

- [“Averiguar lo que contiene el BSDS” en la página 489](#)
- [“Cambiar el BSDS” en la página 491](#)
- [“Recuperación del BSDS” en la página 495](#)

## **z/OS** **Averiguar lo que contiene el BSDS**

Puede utilizar el programa de utilidad imprimir correlación de registros (CSQJU004) para examinar el contenido del BSDS.

El programa de utilidad imprimir correlación de registros (CSQJU004) es un programa de utilidad de proceso por lotes que muestra la información almacenada en el BSDS. Para obtener instrucciones sobre cómo ejecutarlo, consulte [El programa de utilidad de correlación de registro de impresión](#).

El BSDS contiene:

- [Indicaciones de fecha y hora](#)
- [Estado del conjunto de datos de registro activo](#)

### Indicaciones de fecha y hora en el BSDS

La salida del programa de utilidad imprimir correlación de registros muestra las indicaciones de fecha y hora, que se utilizan para registrar la fecha y hora de diversos sucesos del sistema, que están almacenadas en el BSDS.

Las siguientes indicaciones de fecha y hora se incluyen en la sección de cabecera del informe:

#### **SYSTEM TIMESTAMP (INDICACIÓN DE FECHA Y HORA DEL SISTEMA)**

Refleja la fecha y hora de la última actualización del BSDS. La indicación de fecha y hora del BSDS se puede actualizar cuando:

- El gestor de colas se inicia.
- Se alcanza el umbral de grabación durante las actividades de grabación de registro.  
Dependiendo del número de almacenamientos intermedios de salida que haya especificado y la

tasa de actividad del sistema, el BSDS se puede actualizar varias veces por segundo, o puede no actualizarse durante varios segundos, minutos o incluso horas. Para obtener más información sobre el umbral de grabación, consulte el parámetro WRTHRSR de la macro CSQ6LOGP en [Utilización de CSQ6LOGP](#).

- IBM MQ entra en modalidad de BSDS único desde su modalidad de BSDS dual normal debido a un error. Esto puede ocurrir cuando una solicitud para obtener, insertar, apuntar a, actualizar o suprimir un registro BSDS no se realiza satisfactoriamente. Cuando se produce este error, IBM MQ actualiza la indicación de fecha y hora en los BSDS restantes para forzar una discrepancia de indicación de fecha y hora con el BSDS inhabilitado.

#### **UTILITY TIMESTAMP (INDICACIÓN DE FECHA Y HORA DEL PROGRAMA DE UTILIDAD)**

La fecha y hora en que el contenido del BSDS fue alterado por el programa de utilidad de inventario de registro de cambios (CSQJU003).

Las siguientes indicaciones de fecha y hora se incluyen en la parte del informe correspondiente a los conjuntos de datos de registro activo y de archivado:

#### **Fecha del registro activo**

La fecha en que se creó la entrada del registro activo en el BSDS, es decir, cuando se llevó a cabo la función NEWLOG de CSQJU003.

#### **Hora del registro activo**

La hora a la que se creó la entrada del registro activo en el BSDS, es decir, cuando se llevó a cabo la función NEWLOG de CSQJU003.

#### **Fecha del registro de archivado**

La fecha en que se creó la entrada del registro de archivado en el BSDS, es decir, cuando se llevó a cabo la función NEWLOG de CSQJU003 o se llevó a cabo el archivado propiamente dicho.

#### **Hora del registro de archivado**

La hora a la que se creó la entrada del registro de archivado en el BSDS, es decir, cuando se llevó a cabo la función NEWLOG de CSQJU003 o se llevó a cabo el archivado propiamente dicho.

#### **Estado del conjunto de datos de registro activo**

El BSDS registra el estado de un conjunto de datos de registro activo como uno de los siguientes:

##### **NEW**

El conjunto de datos se ha definido pero no ha sido utilizado nunca por IBM MQ o el registro se ha truncado en un punto antes de que el conjunto de datos se utilizara por primera vez. En cualquier caso, los valores de RBA inicial y final del conjunto de datos se restablecen a cero.

##### **REUSABLE (REUTILIZABLE)**

El conjunto de datos se ha definido pero no ha sido utilizado nunca por IBM MQ o el conjunto de datos se ha descargado. En la salida del programa de utilidad imprimir correlación de registros, el valor de RBA inicial para el último conjunto de datos REUSABLE es igual al valor de RBA inicial del último conjunto de datos de registro de archivado.

##### **NOT REUSABLE (NO REUTILIZABLE)**

El conjunto de datos contiene registros que no se han descargado.

##### **STOPPED**

El procesador de descarga ha detectado un error al leer un registro, y ese registro no se ha podido obtener de la otra copia del registro activo.

##### **TRUNCATED (TRUNCADO)**

Realice una de las siguientes acciones:

- Se ha producido un error de E/S y IBM MQ ha dejado de grabar en este conjunto de datos. El conjunto de datos de registro activo se descarga, empezando por la RBA inicial y continuando hasta el último segmento de registro válido en el conjunto de datos de registro activo truncado. La RBA del último segmento de registro válido es menor que la RBA final del conjunto de datos de registro activo. El registro cronológico se cambia al siguiente conjunto de datos de registro activo disponible, y continúa sin interrupción.

o

- Se ha llamado a una función ARCHIVE LOG, que ha truncado el registro activo.

El estado aparece en la salida del programa de utilidad imprimir correlación de registros.

## **Cambiar el BSDS**

No necesita adoptar medidas especiales para mantener el BSDS actualizado con registros de sucesos de registro porque IBM MQ lo hace automáticamente.

Sin embargo, es posible que desee cambiar el BSDS si realiza cualquiera de las siguientes acciones:

- Añadir más conjuntos de datos de registro activo.
- Copiar conjuntos de datos de registro activo en conjuntos de datos recién asignados, por ejemplo, al proporcionar asignaciones de registro activo mayores.
- Mover conjuntos de datos de registro a otros dispositivos.
- Recuperar un BSDS dañado.
- Descartar conjuntos de datos de registro de archivado obsoletos.

Puede cambiar el BSDS ejecutando el programa de utilidad de inventario de registro de cambios (CSQJU003). Ejecute este programa de utilidad sólo cuando el gestor de colas esté inactivo, o puede obtener resultados incoherentes. La acción del programa de utilidad se controla mediante sentencias del conjunto de datos SYSIN. En esta sección se muestran varios ejemplos. Para obtener instrucciones completas, consulte [El programa de utilidad de inventario de registro de cambios](#).

Puede copiar un conjunto de datos de registro activo sólo cuando el gestor de colas está inactivo, ya que IBM MQ asigna los conjuntos de datos de registro activo como exclusivos (DISP=OLD) durante el inicio del gestor de colas.

## **Cambios para registros activos**

Lea este tema para saber cómo puede cambiar los registros activos utilizando el BSDS.

Puede añadir, suprimir y registrar entradas en el BSDS para los registros activos mediante el programa de utilidad cambiar registro. Aquí se muestran sólo ejemplos; sustituya los nombres de conjunto de datos que aparecen por los que desee utilizar. Para obtener más detalles sobre el programa de utilidad, consulte [El programa de utilidad de inventario de registro de cambios](#).

Consulte estas secciones para obtener más información:

- [Añadir entradas de registro al BSDS](#)
- [Suprimir información sobre el conjunto de datos de registro activo del BSDS](#)
- [Registrar información sobre el conjunto de datos de registro en el BSDS](#)
- [Aumentar el tamaño del registro activo](#)
- [El uso de CSQJUFMT](#)

### **Añadir entradas de registro al BSDS**

Si un registro activo se ha marcado como "detenido", no se reutiliza para el registro cronológico; sin embargo, se sigue utilizando para lectura. Utilice los servicios de método de acceso para definir nuevos conjuntos de datos de registro activo, luego utilice el programa de utilidad de inventario de registro de cambios para registrar los nuevos conjuntos de datos en el BSDS. Por ejemplo, utilice:

```
NEWLOG DSNAME=MQM111.LOGCOPY1.DS10,COPY1
NEWLOG DSNAME=MQM111.LOGCOPY2.DS10,COPY2
```

Si va a copiar el contenido de un conjunto de datos de registro activo antiguo en el nuevo, también puede especificar el rango de RBA y las indicaciones de fecha y hora inicial y final en la función NEWLOG.

## Suprimir información sobre el conjunto de datos de registro activo del BSDS

Para suprimir información sobre un conjunto de datos de registro activo del BSDS, puede utilizar:

```
DELETE DSNNAME=MQM111.LOGCOPY1.DS99
DELETE DSNNAME=MQM111.LOGCOPY2.DS99
```

## Registrar información sobre el conjunto de datos de registro en el BSDS

Para registrar información sobre un conjunto de datos de registro activo existente en el BSDS, utilice:

```
NEWLOG DSNNAME=MQM111.LOGCOPY1.DS10,COPY2,STARTIME=19930212205198,
ENDTIME=19930412205200,STARTRBA=6400,ENDRBA=94FF
```

Es posible que tenga que insertar un registro que contenga este tipo de información en el BSDS porque:

- La entrada para el conjunto de datos se ha suprimido, pero se necesita de nuevo.
- Va a copiar el contenido de un conjunto de datos de registro activo en otro conjunto de datos.
- Va a recuperar el BSDS desde una copia de seguridad.

## Aumentar el tamaño del registro activo

Hay dos métodos de obtener este proceso.

1. Cuando el gestor de colas está activo:
  - a. Defina nuevos conjuntos de datos de registro de gran tamaño utilizando JCL.
  - b. Añada los nuevos conjuntos de datos al gestor de colas activo mediante el mandato MQSC DEFINE LOG.
  - c. Utilice el mandato MQSC ARCHIVE LOG para mover el registro activo actual, de modo que sea un registro nuevo más grande.
  - d. Espere a que se complete el archivado del conjunto de datos del registro más pequeño.
  - e. Cierre el gestor de colas usando la utilidad CSQJU003 para eliminar los registros activos pequeños antiguos.
  - f. Reinicie el gestor de colas.
2. Cuando el gestor de colas está inactivo:
  - a. Detenga el gestor de colas. Este paso es necesario porque IBM MQ asigna todos los conjuntos de datos de registro activo para su uso exclusivo cuando está activo.
  - b. Utilice el mandato AMS ALTER con la opción NEWNAME para renombrar los conjuntos de datos de registro activo.
  - c. Utilice el mandato AMS DEFINE para definir conjuntos de datos de registro activo mayores.

Al reutilizar los nombres de conjunto de datos antiguos, no necesita ejecutar el programa de utilidad de inventario de registro de cambios para establecer nombres nuevos en los BSDS. Los nombres de conjunto de datos antiguos y los rangos de RBA correctos ya están en los BSDS.
  - d. Utilice el mandato AMS REPRO para copiar los conjuntos de datos antiguos (renombrados) en sus respectivos conjuntos de datos nuevos.

**Nota:** Este paso puede tardar mucho tiempo, por lo que su empresa podría quedar inactiva durante este periodo.
  - e. Inicie el gestor de colas.

Si todos los conjuntos de datos de registro son del mismo tamaño, el sistema funcionará de forma más coherente y eficiente. Si los conjuntos de datos de registro no son del mismo tamaño, es más difícil hacer un seguimiento de los registros del sistema, por lo que puede desperdiciarse espacio.

### El uso de CSQJUFMT

No ejecute un formato CSQJUFMT al aumentar el tamaño de un registro activo.

Si ejecuta CSQJUFMT (para proporcionar una ventaja sobre el rendimiento la primera vez que el gestor de colas graba en el nuevo registro activo) recibe los mensajes:

```
IEC070I 203-204,XS95GLX,REPRO02,OUTPUT,B857,SPMG02, 358
IEC070I MG.W.MG4E.LOGCOPY1.DS02,MG.W.MG4E.LOGCOPY1.DS02.DATA,
IDC3302I ACTION ERROR ON MG.W.MG4E.LOGCOPY1.DS02
IDC3351I ** VSAM I/O RETURN CODE IS 28 - RPLFDBWD = X'2908001C'
IDC31467I MAXIMUM ERROR LIMIT REACHED.

IDC0005I NUMBER OF RECORDS PROCESSED WAS 0
```

Además, si utiliza REPRO de servicios de método de acceso, asegúrese de definir un nuevo registro vacío.

Si utiliza REPRO para copiar el conjunto de datos antiguo (renombrado) en su respectivo conjunto de datos nuevo, el valor predeterminado es NOREPLACE.

Esto significa que REPRO no sustituye un registro que ya está en el conjunto de datos designado. Cuando se lleva a cabo el formato en el conjunto de datos, el valor de RBA se restablece. El resultado neto es un conjunto de datos que no está vacío después del formateo.

### Cambios para registros de archivado

Lea este tema para informarse sobre cómo cambiar los registros de archivado.

Puede añadir, suprimir y cambiar la contraseña de entradas en el BSDS para registros de archivado. Aquí se muestran sólo ejemplos; sustituya los nombres de conjunto de datos que aparecen por los que desee utilizar. Para obtener más detalles sobre el programa de utilidad, consulte [El programa de utilidad de inventario de registro de cambios](#).

- [Añadir un registro de archivado](#)
- [Suprimir un registro de archivado](#)
- [Cambiar la contraseña de un registro de archivado](#)

### Añadir un registro de archivado

Cuando la recuperación de un objeto depende de la lectura de un conjunto de datos de registro de archivado existente, el BSDS debe contener información sobre ese conjunto de datos para que IBM MQ pueda encontrarlo. Para registrar información sobre un conjunto de datos de registro de archivado existente en el BSDS, utilice:

```
NEWLOG DSNAME=CSQARC1.ARCHLOG1.E00021.T2205197.A0000015,COPY1VOL=CSQV04,
UNIT=TAPE,STARTRBA=3A190000,ENDRBA=3A1F0FFF,CATALOG=NO
```

### Suprimir un registro de archivado

Para suprimir un conjunto de datos de registro de archivado completo en uno o varios volúmenes, utilice:

```
DELETE DSNAME=CSQARC1.ARCHLOG1.E00021.T2205197.A0000015,COPY1VOL=CSQV04
```

## Cambiar la contraseña de un registro de archivado

Si cambia la contraseña de un conjunto de datos de registro de archivado existente, también debe cambiar la información en el BSDS.

1. Liste el BSDS, utilizando el programa de utilidad imprimir correlación de registros.
2. Suprima la entrada para el conjunto de datos de registro de archivado con la contraseña cambiada, utilizando la función DELETE del programa de utilidad CSQJU003 (consulte el tema [El programa de utilidad de inventario de registro de cambios](#)).
3. Asigne un nombre al conjunto de datos igual que para un nuevo conjunto de datos de registro de archivado. Utilice la función NEWLOG del programa de utilidad CSQJU003 (consulte el tema [El programa de utilidad de inventario de registro de cambios](#)) y proporcione la nueva contraseña, las RBA de inicio y final y los números de serie de volumen (que se pueden encontrar en la salida del programa de utilidad de correlación de registro de impresión, consulte [El programa de utilidad de correlación de registro de impresión](#)).

Para cambiar la contraseña de nuevos conjuntos de datos de registro de archivado, utilice:

```
ARCHIVE PASSWORD= password
```

Para dejar de establecer contraseñas en nuevos conjuntos de datos de registro de archivado, utilice:

```
ARCHIVE NOPASSWD
```

**Nota:** Utilice solamente la función de programa de utilidad ARCHIVE si no tiene un gestor de seguridad externo.

## z/OS Cambiar el calificador de alto nivel (HLQ) para los registros y el BSDS

Lea este tema para conocer el procedimiento necesario para cambiar el calificador de alto nivel (HLQ).

### Antes de empezar

Debe finalizar el gestor de colas normalmente antes de copiar cualquiera de los registros o conjuntos de datos en los nuevos conjuntos de datos. Esto es para asegurar que los datos sean coherentes y no sea necesaria ninguna recuperación durante el reinicio.

### Acerca de esta tarea

Esta tarea proporciona información sobre cómo cambiar el HLQ para los registros y el BSDS. Para ello, siga estos pasos:

### Procedimiento

1. Ejecute el programa de utilidad imprimir registro CSQJU004 para registrar la información del conjunto de datos de registro. Esta información la necesitará más tarde.
2. Puede efectuar una de las acciones siguientes:
  - a) ejecutar la copia de seguridad y restauración DSS con renombrar en los conjuntos de datos de registro y BSDS que se han de renombrar, o
  - b) utilizar los mandatos AMS DEFINE y REPRO para crear los conjuntos de datos HLQ y copiar los datos de los conjuntos de datos antiguos.
3. Modifique los procedimientos MSTR y CHIN para que apunten a los nuevos conjuntos de datos.
4. Suprima la información de registro antigua en la nueva copia de BSDS utilizando el programa de utilidad CSQJU003.

5. Defina los nuevos conjuntos de datos de registro en el nuevo BSDS utilizando la función NEWLOG de CSQJU003.

Mantenga igual toda la información acerca de cada registro, separada de HLQ.

6. El nuevo BSDS debe reflejar la misma información que se registró para los registros antiguos en el BSDS antiguo.

El HLQ debe ser lo único que sea diferente.

## Qué hacer a continuación

Compare la salida de CSQJU004 para el BSDS antiguo y nuevo para asegurarse de que son EXACTAMENTE igual (excepto el HLQ) antes de iniciar el gestor de colas.

**Nota:** Hay que tener cuidado al realizar estas operaciones. Acciones incorrectas pueden conducir a situaciones irrecuperables. Compruebe la salida del programa de utilidad PRINT LOG MAP y asegúrese de que se ha incluido toda la información necesaria para la recuperación o el reinicio.

## Recuperación del BSDS

Si IBM MQ está funcionando en modalidad de BSDS dual y un BSDS queda dañado, lo que fuerza a IBM MQ a pasar a la modalidad de BSDS único, IBM MQ sigue funcionando sin problemas (hasta el próximo reinicio).

Para devolver el entorno a la modalidad de BSDS dual:

1. Utilice los Servicios de método de acceso para renombrar o suprimir el BSDS dañado y para definir un nuevo BSDS con el mismo nombre que el BSDS dañado. Puede encontrar sentencias de control de ejemplo en el trabajo CSQ4BREC en thlqual.SCSQPROC.
2. Emita el mandato de IBM MQ RECOVER BSDS para hacer una copia del BSDS válido en el conjunto de datos recién asignado y para restablecer la modalidad de BSDS dual.

Si IBM MQ está funcionando en modalidad de BSDS único y el BSDS queda dañado, o si IBM MQ está funcionando en modalidad de BSDS dual y los dos BSDS quedan dañados, el gestor de colas se detiene y no se reinicia hasta que los conjuntos de datos BSDS se han reparado. En este caso:

1. Localice el BSDS asociado al conjunto de datos de registro de archivado más reciente. El nombre de conjunto de datos del registro de archivado más reciente aparece en el registro de trabajo en la última aparición del mensaje CSQJ003I, que indica que el proceso de descarga se ha completado satisfactoriamente. En preparación para el resto de este procedimiento, es una buena práctica mantener un registro de todos los archivados satisfactorios indicados por ese mensaje:
  - Si los registros de archivado están en DASD, el BSDS se asigna en cualquier DASD disponible. El nombre de BSDS es igual que el nombre del conjunto de datos de registro de archivado correspondiente; cambie sólo la primera letra del último calificador, de A a B, como en este ejemplo:

### Nombre del registro de archivado

CSQ.ARCHLOG1. **A** 0000001

### Nombre de la copia BSDS

CSQ.ARCHLOG1. **B** 0000001

- Si los registros de archivado están en cinta, el BSDS es el primer conjunto de datos del primer volumen de registro de archivado. El BSDS no se repite en volúmenes posteriores.
2. Si el conjunto de datos de registro de archivado más reciente no tiene ninguna copia del BSDS (por ejemplo, porque se ha producido un error al descargarlo), localice una copia anterior del BSDS de un proceso de descarga anterior.
  3. Renombre los BSDS *dañados* mediante el mandato ALTER de los Servicios de método de acceso con la opción NEWNAME. Si desea suprimir un BSDS dañado, utilice el mandato DELETE de los Servicios de método de acceso. Para cada BSDS dañado, utilice los Servicios de método de acceso (AMS) para definir un nuevo BSDS como un conjunto de datos de sustitución. El trabajo CSQ4BREC en thlqual.SCSQPROC contiene sentencias de control de los Servicios de método de acceso para definir un nuevo BSDS.

4. Utilice el mandato REPRO de los Servicios de método de acceso para copiar el BSDS del registro de archivado en uno de los BSDS de sustitución que ha definido en el paso “3” en la página 495. No copie ningún dato en el segundo BSDS de sustitución, lo hará en el paso “5” en la página 497.

a. Imprima el contenido del BSDS de sustitución.

Utilice el programa de utilidad imprimir correlación de registros (CSQJU004) para imprimir el contenido del BSDS de sustitución. Esto le permite revisar el contenido del BSDS de sustitución antes de continuar con el trabajo de recuperación.

b. Actualice el inventario de conjuntos de datos de registro de archivado en el BSDS de sustitución.

Examine la salida del programa de utilidad imprimir correlación de registros y compruebe que el BSDS de sustitución no contiene un registro del registro de archivado desde el que se ha copiado el BSDS. Si el BSDS de sustitución es una copia antigua, el inventario podría no contener todos los conjuntos de datos de registro de archivado que se han creado más recientemente. El inventario de BSDS de los conjuntos de datos de registro de archivado debe actualizarse para que refleje el inventario actual del subsistema.

Utilice la sentencia NEWLOG del programa de utilidad de inventario de registro de cambios (CSQJU003) para actualizar el BSDS de sustitución, añadiendo un registro del registro de archivado desde el que se ha copiado el BSDS. Si el conjunto de datos de registro de archivado está protegido por contraseña, utilice la opción PASSWORD de la función NEWLOG. Asimismo, si el conjunto de datos de registro de archivado está catalogado, asegúrese de que la opción CATALOG de la función NEWLOG está establecida correctamente en CATALOG=YES. Utilice la sentencia NEWLOG para añadir los conjuntos de datos de registro de archivado adicionales que se crearon con posterioridad a la copia del BSDS.

c. Actualice las contraseñas en el BSDS de sustitución.

El BSDS contiene contraseñas para los conjuntos de datos de registro de archivado y para los conjuntos de datos de registro activo. Para asegurarse de que las contraseñas en el BSDS de sustitución reflejan las contraseñas actuales utilizadas por su instalación, utilice la función ARCHIVE del programa de utilidad de inventario de registro de cambios con la opción PASSWORD.

d. Actualice el inventario de conjuntos de datos de registro activo en el BSDS de sustitución.

En circunstancias excepcionales, la instalación podría haber añadido, suprimido o renombrado conjuntos de datos de registro activo desde que se copió el BSDS. En este caso, el BSDS de sustitución no refleja el número real o los nombres de los conjuntos de datos de registro activo que la instalación tiene actualmente en uso.

Si necesita suprimir un conjunto de datos de registro activo del inventario de registros del BSDS de sustitución, utilice la función DELETE del programa de utilidad de inventario de registro de cambios.

Si necesita añadir un conjunto de datos de registro activo al inventario de registros del BSDS de sustitución, utilice la función NEWLOG del programa de utilidad de inventario de registro de cambios. Asegúrese de que el rango de RBA se ha especificado correctamente en la función NEWLOG. Si el conjunto de datos de registro activo está protegido por contraseña, utilice la opción PASSWORD.

Si tiene que renombrar un conjunto de datos de registro activo en el inventario de registros del BSDS de sustitución, utilice la función DELETE del programa de utilidad de inventario de registro de cambios, seguida de la función NEWLOG. Asegúrese de que el rango de RBA se ha especificado correctamente en la función NEWLOG. Si el conjunto de datos de registro activo está protegido por contraseña, utilice la opción PASSWORD.

e. Actualice los rangos de RBA de registro activo en el BSDS de sustitución.

Posteriormente, cuando el gestor de colas se reinicia, compara las RBA de los conjuntos de datos de registro activo listados en el BSDS con las RBA que se han encontrado en los conjuntos de datos de registro activo reales. Si las RBA no coinciden, el gestor de colas no se reinicia. El problema se magnifica cuando se utiliza una copia antigua del BSDS. Para solucionar este problema, utilice el programa de utilidad de inventario de registro de cambios (CSQJU003) para ajustar las RBA



encontradas en el BSDS con las RBA en los conjuntos de datos de registro activo reales. Esto se hace:

- Utilizando el programa de utilidad imprimir registros de anotaciones (CSQ1LOGP) para imprimir un informe de resumen del conjunto de datos de registro activo. Esto muestra las RBA inicial y final.
- Comparando los rangos de RBA reales con los rangos de RBA que acaba de imprimir, cuando se conocen las RBA de todos los conjuntos de datos de registro activo.

Si los rangos de RBA son iguales para todos los conjuntos de datos de registro activo, puede continuar con el siguiente paso de recuperación sin realizar ningún trabajo adicional.

Si los rangos de RBA no son iguales, ajuste los valores en el BSDS para reflejar los valores reales. Para cada conjunto de datos de registro activo en el que se debe ajustar el rango de RBA, utilice la función DELETE del programa de utilidad de inventario de registro de cambios para suprimir el conjunto de datos de registro activo del inventario en el BSDS de sustitución. A continuación, utilice la función NEWLOG para redefinir el conjunto de datos de registro activo en el BSDS. Si los conjuntos de datos de registro activo están protegidos por contraseña, utilice la opción PASSWORD de la función NEWLOG.

- f. Si sólo se especifican dos conjuntos de datos de registro activo para cada copia del registro activo, IBM MQ puede tener dificultades durante el reinicio del gestor de colas. El problema puede surgir cuando uno de los conjuntos de datos de registro activo está lleno y no se ha descargado, mientras que el segundo conjunto de datos de registro activo está a punto de llenarse. En este caso, añada un nuevo conjunto de datos de registro activo para cada copia del registro activo y defina cada nuevo conjunto de datos de registro activo en el inventario de registros del BSDS de sustitución.

Utilice el mandato DEFINE de los Servicios de método de acceso para definir un nuevo conjunto de datos de registro activo para cada copia del registro activo, y utilice la función NEWLOG del programa de utilidad de inventario de registro de cambios para definir los nuevos conjuntos de datos de registro activo en el BSDS de sustitución. No es necesario especificar los rangos de RBA en la sentencia NEWLOG. Sin embargo, si los conjuntos de datos de registro activo están protegidos por contraseña, utilice la opción PASSWORD de la función NEWLOG. Puede encontrar sentencias de control de ejemplo para realizar esta tarea en el trabajo CSQ4LREC en thlqual.SCSQPROC.

5. Copie el BSDS actualizado en el segundo conjunto de datos BSDS nuevo. Los BSDS ahora son idénticos.

Utilice el programa de utilidad imprimir correlación de registros (CSQJU004) para imprimir el contenido del segundo BSDS de sustitución en este punto.

6. Consulte [Problemas de registros activos](#) para obtener información sobre qué hacer si ha perdido su conjunto de datos de registro activo actual.
7. Reinicie el gestor de colas utilizando el BSDS que acaba de construir. IBM MQ determina la RBA actual y qué registros activos deben archivar.

## **Gestión de conjuntos de páginas**

Utilice este tema para comprender cómo gestionar los conjuntos de páginas asociados a un gestor de colas.

Este tema describe cómo añadir, copiar y gestionar de forma general los conjuntos de páginas asociados a un gestor de colas. Contiene las siguientes secciones:

- [“Cómo cambiar el calificador de alto nivel \(HLQ\) para los conjuntos de páginas”](#) en la página 498
- [“Cómo añadir un conjunto de páginas a un gestor de colas”](#) en la página 498
- [“Qué hace cuando uno de los conjuntos de páginas se llena”](#) en la página 499
- [“Cómo equilibrar las cargas en los conjuntos de páginas”](#) en la página 499

- [Cómo aumentar el tamaño de un conjunto de páginas](#)
- [“Cómo reducir un conjunto de páginas” en la página 503](#)
- [“Cómo reintroducir un conjunto de páginas” en la página 503](#)
- [“Cómo hacer una copia de seguridad y recuperar conjuntos de páginas” en la página 504](#)
- [“Cómo suprimir conjuntos de páginas” en la página 508](#)
- [“Cómo hacer una copia de seguridad y restaurar las colas utilizando CSQUTIL” en la página 508](#)

Consulte [Conjuntos de páginas](#) para obtener una descripción de conjuntos de páginas, clases de almacenamiento, almacenamientos intermedios y agrupaciones de almacenamiento intermedio y algunas de las consideraciones de rendimiento que se aplican.

## **Cómo cambiar el calificador de alto nivel (HLQ) para los conjuntos de páginas**

Esta tarea proporciona información sobre cómo cambiar el HLQ para los conjuntos de páginas. Para realizar esta tarea, haga lo siguiente:

1. Defina los nuevos conjuntos de páginas del HLQ.
2. Si la asignación de tamaño es la misma que la de los conjuntos de páginas antiguos, copie el conjunto de páginas existente utilizando REPRO para los nuevos conjuntos de páginas HLQ vacíos.
3. Si está aumentando el tamaño de los conjuntos de páginas, utilice la función FORMAT de CSQUTIL para formatear las páginas de destino y, a continuación, la función COPYPAGE de CSQUTIL para copiar todos los mensajes del conjunto de páginas de origen al conjunto de páginas de destino.  
  
Para obtener más información, consulte [Formato de conjuntos de páginas \(FORMAT\)](#) y [Expansión de un conjunto de páginas \(COPYPAGE\)](#).
4. Cambie la sentencia CSQP00xx DD en el procedimiento del gestor de colas para apuntar a los nuevos conjuntos de páginas HLQ.

Reinicie el gestor de colas y verifique los cambios en los conjuntos de páginas.

## **Cómo añadir un conjunto de páginas a un gestor de colas**

Esta descripción asume que tiene un gestor de colas que ya está en ejecución. Es posible que tenga que añadir un conjunto de páginas si, por ejemplo, el gestor de colas tiene que hacer frente a nuevas aplicaciones utilizando nuevas colas.

Para añadir un nuevo conjunto de páginas, utilice el siguiente procedimiento:

1. Defina y formatee el nuevo conjunto de páginas. Puede utilizar el JCL de ejemplo en [thlqual.SCSQPROC\(CSQ4PAGE\)](#) como base. Para obtener más información, consulte [Dar formato a conjuntos de páginas \(FORMAT\)](#).  
  
Tenga cuidado en no formatear ningún conjunto de páginas que esté siendo utilizado, a menos que sea esto lo que pretende. Si es así, utilice la opción FORCE de la función de programa de utilidad FORMAT.
2. Utilice el mandato DEFINE PSID con la opción DSN para asociar el conjunto de páginas a una agrupación de almacenamiento intermedio.
3. Añada las definiciones de la clase de almacenamiento apropiadas para el conjunto de páginas emitiendo mandatos DEFINE STGCLASS.
4. Opcionalmente, para documentar cómo se ha configurado su gestor de colas:
  - a. Añada el nuevo conjunto de páginas al procedimiento de la tarea iniciado para el gestor de colas.
  - b. Añada una definición al nuevo conjunto de páginas al conjunto de datos de inicialización de CSQINP1.
  - c. Añada una definición a la nueva clase de almacenamiento en su miembro de conjunto de datos de inicialización de CSQ4INYP.

Si desea ver detalles de los mandatos DEFINE PSID y DEFINE STGCLASS, consulte [DEFINE PSID](#) y [DEFINE STGCLASS](#).

## Qué hace cuando uno de los conjuntos de páginas se llena

Puede averiguar el uso de los conjuntos de páginas mediante el mandato de IBM MQ DISPLAY USAGE. Por ejemplo, el mandato:

```
DISPLAY USAGE PSID(03)
```

muestra el estado actual del conjunto de páginas 03. Esto le indica cuántas páginas libres tiene este conjunto de páginas.

Si ha definido ampliaciones secundarias para los conjuntos de páginas, se amplían dinámicamente cada vez que se llenan. Finalmente, se utilizan todas las ampliaciones secundarias o no hay disponible espacio en disco adicional. Si esto sucede, una aplicación recibe el código de retorno MQRC\_STORAGE\_MEDIUM\_FULL.

Si una aplicación recibe un código de retorno de MQRC\_STORAGE\_MEDIUM\_FULL de una llamada de MQI, esto es una indicación clara de que no queda suficiente espacio en el conjunto de páginas. Si el problema persiste o es probable que vuelva a ocurrir, debe hacer algo para resolverlo.

Puede enfocar este problema de varias maneras:

- Equilibrar la carga entre conjuntos de páginas moviendo colas de un conjunto de páginas a otro.
- Ampliar el conjunto de páginas. Consulte [“Cómo aumentar el tamaño de un conjunto de páginas”](#) en la [página 501](#) para obtener instrucciones.
- Redefina el conjunto de páginas para que se pueda ampliar más allá de 4 GB hasta un tamaño máximo de 64 GB. Consulte [Definición de un conjunto de páginas para que tenga un tamaño superior a 4 GB](#) para obtener instrucciones.

## Cómo equilibrar las cargas en los conjuntos de páginas

El equilibrado de carga en conjuntos de páginas significa mover los mensajes asociados a una o más colas de un conjunto de páginas a otro, menos utilizado. Utilice esta técnica si no es práctico ampliar el conjunto de páginas.

Para identificar qué colas están utilizando un conjunto de páginas, utilice los mandatos apropiados de IBM MQ. Por ejemplo, para descubrir qué colas están correlacionadas con el conjunto de páginas 02, en primer lugar, averigüe qué clases de almacenamiento se correlacionan con el conjunto de páginas 02, mediante el mandato:

```
DISPLAY STGCLASS(*) PSID(02)
```

Después, utilice el siguiente mandato para descubrir qué colas utilizan la clase de almacenamiento:

```
DISPLAY QUEUE(*) TYPE(QLOCAL) STGCLASS
```

## Traslado a una cola no compartida

Para mover colas y sus mensajes de un conjunto de páginas a otro, utilice el mandato MQSC MOVE QLOCAL (descrito en [MOVE QLOCAL](#)). Cuando haya identificado la cola o las colas que desea mover a un nuevo conjunto de páginas, siga este procedimiento para cada una de estas colas:

1. Asegúrese de que la cola que desea mover no está siendo utilizada por ninguna aplicación (es decir, los valores IPPROCS y OPPROCS del mandato DISPLAY QSTATUS son cero) y que no tiene mensajes no confirmados (el valor UNCOM del mandato DISPLAY QSTATUS es NO).

**Nota:** La única forma de asegurarse de que este estado continúa es cambiar temporalmente la autorización de seguridad de la cola. Consulte [Perfiles para la seguridad de colas](#) para obtener más información.

Si no puede hacer esto, las etapas posteriores de este procedimiento podrían fallar si las aplicaciones empiezan a utilizar la cola, a pesar de las medidas cautelares como, por ejemplo, definir el valor PUT(INHABILITADO). Sin embargo, los mensajes nunca se pueden perder mediante este procedimiento.

2. Impida que las aplicaciones pongan mensajes en la cola que se está moviendo alterando la definición de cola para inhabilitar MQPUT. Cambie la definición de cola por PUT (DISABLED).
3. Defina una cola temporal con los mismos atributos que la cola que se está trasladando, mediante el mandato:

```
DEFINE QL(TEMP_QUEUE) LIKE(Queue_To_Move) PUT(ENABLED) GET(ENABLED)
```

**Nota:** Si esta cola temporal ya existe de una ejecución anterior, suprimala antes de realizar la definición.

4. Mueva los mensajes a la cola temporal mediante el mandato siguiente:

```
MOVE QLOCAL(Queue_To_Move) TOQLOCAL(TEMP_QUEUE)
```

5. Suprima la cola que está moviendo, mediante el mandato:

```
DELETE QLOCAL(Queue_To_Move)
```

6. Defina una nueva clase de almacenamiento que se correlaciona con el conjunto de páginas necesario, por ejemplo:

```
DEFINE STGCLASS(NEW) PSID(nn)
```

Añada la nueva definición de la clase de almacenamiento a los conjuntos de datos CSQINP2 preparados para el siguiente reinicio del gestor de colas.

7. Redefina la cola que está moviendo, modificando el atributo de la clase de almacenamiento:

```
DEFINE QL(Queue_To_Move) LIKE(TEMP_QUEUE) STGCLASS(NEW)
```

Cuando se redefine la cola, ésta se basa en la cola temporal que se ha creado en el paso [“3”](#) en la [página 500](#).

8. Vuelva a mover los mensajes a la nueva cola mediante el mandato:

```
MOVE QLOCAL(TEMP) TOQLOCAL(Queue_To_Move)
```

9. La cola creada en el paso “3” en la página 500 ya no es necesaria. Utilice el mandato siguiente para suprimirla:

```
DELETE QL(TEMP_QUEUE)
```

10. Si la cola que se está moviendo se definió en los conjuntos de datos CSQINP2, cambie el atributo STGCLASS del mandato DEFINE QLOCAL apropiado en los conjuntos de datos CSQINP2. Añada la palabra clave REPLACE de modo que se sustituya la definición de cola existente.

La Figura 40 en la página 501 muestra un ejemplo de un trabajo de equilibrio de carga.

```
//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=thlqual.SCSQANLE,DISP=SHR
//      DD DSN=thlqual.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
ALTER QL(Queue_To_Move) PUT(DISABLED)
DELETE QL(TEMP_QUEUE) PURGE
DEFINE QL(TEMP_QUEUE) LIKE(Queue_To_Move) PUT(ENABLED) GET(ENABLED)
MOVE QLOCAL(Queue_To_Move) TOQLOCAL(TEMP_QUEUE)
DELETE QL(Queue_To_Move)
DEFINE STGCLASS(NEW) PSID(2)
DEFINE QL(Queue_To_Move) LIKE(TEMP_QUEUE) STGCLASS(NEW)
MOVE QLOCAL(TEMP_QUEUE) TOQLOCAL(Queue_To_Move)
DELETE QL(TEMP_QUEUE)
/*
```

Figura 40. Extracto de un trabajo de equilibrio de carga para un conjunto de páginas

## Cómo aumentar el tamaño de un conjunto de páginas

Puede asignar de forma inicial un conjunto de páginas mayor que 4 GB, consulte [Definición de un conjunto de páginas para que sea mayor que 4 GB](#)

Un conjunto de páginas se puede definir para que se expanda automáticamente a medida que se llena especificando EXPAND (SYSTEM) o EXPAND (USER). Si el conjunto de páginas se definió con EXPAND(NINGUNO), puede ampliarlo de una de estas dos formas:

- Alterar su definición para permitir la ampliación automática. Consulte [Modificación de un conjunto de páginas para permitir la ampliación automática](#)
- Crear un nuevo conjunto de páginas mayor y copiar los mensajes del conjunto de páginas antiguo al nuevo. Consulte [Traslado de mensajes a un nuevo conjunto de páginas mayor](#)

### Definición de un conjunto de páginas para que sea mayor que 4 GB

IBM MQ puede utilizar un conjunto de páginas de hasta 64 GB de tamaño, siempre que el conjunto de páginas esté definido con la 'direccionabilidad ampliada' en VSAM. La direccionabilidad ampliada es un atributo que concede una clase de datos SMS.

**Nota:** Los conjuntos de páginas y los conjuntos de datos de registro activos son elegibles para residir en la parte de espacio de direccionamiento ampliado (EAS) de un volumen de direcciones ampliado (EAV) y, desde z/OS V1.12, un conjunto de datos de registro de archivado también puede residir en el EAS.

En el ejemplo que se muestra en el siguiente JCL de ejemplo, la clase de gestión 'EXTENDED' se define en SMS con la 'direccionabilidad ampliada'. Si el conjunto de páginas existente no está definido actualmente como que tiene una direccionabilidad ampliada, utilice el método siguiente para migrar a un conjunto de datos de formato de direccionabilidad ampliada.

1. Detenga el gestor de colas.
2. Utilice los servicios de método de acceso para cambiar el nombre del conjunto de páginas existente.
3. Defina un conjunto de páginas de destino, el mismo tamaño que el conjunto de páginas existente, pero con DATACLAS(AMPLIADO).

**Nota:** Los conjuntos de datos con formato ampliado deben poder ser gestionados con SMS. Existen los mecanismos para solicitar un formato ampliado para conjuntos de datos VSAM:

- Utilización de una clase de datos que tenga un valor DSNTYPE de EXT y el subparámetro R o P para indicar necesario o preferido.
- Codificación de DSNTYPE=EXTREQ (el formato ampliado es necesario) o DSNTYPE=EXTPREF (el formato ampliado es preferido) en la sentencia DD.
- Codificación del parámetro LIKE= en la sentencia DD para hacer referencia a un conjunto de datos de formato ampliado existente.

Para obtener más información, consulte [Restricciones en la definición de conjuntos de datos de formato ampliado](#).

4. Utilice la función COPYPAGE de CSQUTIL para copiar todos los mensajes del conjunto de páginas de origen al conjunto de páginas de destino. Consulte [Ampliación de un conjunto de páginas \(COPYPAGE\)](#) para obtener más detalles.
5. Reinicie el gestor de colas.
6. Modifique el conjunto de páginas para utilizar la ampliación del sistema, para permitir que siga creciendo más allá de su asignación actual.

El siguiente JCL muestra mandatos de ejemplo de los servicios de método de acceso:

```
//S1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
ALTER 'VICY.CSQ1.PAGE01' -
NEWNAME('VICY.CSQ1.PAGE01.OLD')
ALTER 'VICY.CSQ1.PAGE01.DATA' -
NEWNAME('VICY.CSQ1.PAGE01.DATA.OLD')
DEFINE CLUSTER (NAME('VICY.CSQ1.PAGE01') -
MODEL('VICY.CSQ1.PAGE01.OLD') -
DATACLAS(EXTENDED))
/*
```

### Modificación de un conjunto de páginas para permitir la ampliación automática

Utilice el mandato ALTER PSID con las opciones EXPAND(USUARIO) o EXPAND(SISTEMA). Consulte [ALTER PSID](#) y [de un conjunto de páginas \(COPYPAGE\)](#) para obtener información general sobre la ampliación de conjuntos de páginas.

### Traslado de mensajes a un nuevo conjunto de páginas mayor

Esta técnica implica detener y reiniciar el gestor de colas. Así se suprimen los mensajes no persistentes que no están en las colas compartidas durante el reinicio. Si tiene mensajes no persistentes que no desea suprimir, utilice el equilibrio de carga en su lugar. Para obtener más detalles, consulte el apartado [“Cómo equilibrar las cargas en los conjuntos de páginas”](#) en la [página 499](#). En esta descripción, se hace referencia al conjunto de páginas que desea ampliar como el conjunto de páginas *de origen*; se hace referencia al nuevo conjunto de páginas mayor como el conjunto de páginas *de destino*.

Siga estos pasos:

1. Detenga el gestor de colas.
2. Defina el conjunto de páginas de destino, asegurándose de que es mayor que el conjunto de páginas de origen, con un valor de ampliación secundaria mayor.

3. Utilice la función `FORMAT` de `CSQUTIL` para dar formato al conjunto de páginas de destino. Consulte [Dar formato a conjuntos de páginas \(FORMAT\)](#) para obtener más información.
4. Utilice la función `COPYPAGE` de `CSQUTIL` para copiar todos los mensajes del conjunto de páginas de origen al conjunto de páginas de destino. Consulte [Ampliación de un conjunto de páginas \(COPYPAGE\)](#) para obtener más detalles.
5. Reinicie el gestor de colas utilizando el conjunto de páginas de destino haciendo una de las acciones siguientes:
  - Cambie el procedimiento de la tarea iniciada del gestor de colas para hacer referencia al conjunto de páginas de destino.
  - Utilice los servicios de método de acceso para suprimir el conjunto de páginas de origen y, a continuación, cambie el nombre del conjunto de páginas de destino, lo que le proporciona el mismo nombre que el del conjunto de páginas de origen.

**Atención:**

Antes de suprimir cualquier conjunto de páginas de IBM MQ, asegúrese de que ha realizado las copias de seguridad necesarias.

## Cómo reducir un conjunto de páginas

Impida a todos los usuarios, que no sean el administrador de IBM MQ, que utilicen el gestor de colas. Por ejemplo; modificando los valores de seguridad de acceso.

Si tiene un conjunto de páginas grande que esté casi vacío (tal como indica el mandato `DISPLAY USAGE`), es posible que desee reducir su tamaño. El procedimiento para hacer esto implica utilizar las funciones `COPY`, `FORMAT` y `LOAD` de `CSQUTIL` (consulte [Programa de utilidad de IBM MQ](#)). Este procedimiento no funciona para el conjunto de páginas cero (0), ya que no es práctico para reducir el tamaño de este conjunto de páginas; la única forma de hacerlo es reiniciando el gestor de colas (consulte ["Reinicialización de un gestor de colas"](#) en la página 529). El requisito previo de este procedimiento es intentar y eliminar todos los usuarios del sistema, de modo que todos los UOW se completan y los conjuntos de páginas son coherentes.

1. Utilice el mandato `STOP QMGR` con la opción `QUIESCE` o `FORCE` para detener el gestor de colas.
2. Ejecute la función `SCOPY` de `CSQUTIL` con la opción `PSID` para copiar todos los mensajes del conjunto de páginas grande y guardarlos en un conjunto de datos secuenciales.
3. Defina un nuevo conjunto de datos de conjunto de páginas más pequeño para sustituir el conjunto de datos grande.
4. Ejecute la función `FORMAT TYPE(NEW)` de `CSQUTIL` en el conjunto de páginas que ha creado en el paso "3" en la página 503.
5. Reinicie el gestor de colas utilizando el conjunto de páginas creado en el paso "3" en la página 503.
6. Ejecute la función `LOAD` de `CSQUTIL` para volver a cargar todos los mensajes guardados durante el paso "2" en la página 503.
7. Permita a todos los usuarios acceder al gestor de colas.
8. Suprima el conjunto de páginas grande antiguo.

## Cómo reintroducir un conjunto de páginas

En determinados escenarios, es útil poder volver a activar un conjunto de páginas antiguo en el gestor de colas. A menos que se lleve a cabo una acción específica, cuando el conjunto de páginas antiguo se pone en línea, el gestor de colas reconocerá que el RBA de recuperación del conjunto de páginas RBA almacenado en el propio conjunto de páginas y en los registros del punto de comprobación es antiguo y, por lo tanto, iniciará automáticamente la recuperación del soporte del conjunto de páginas para actualizarlo.

Dicha recuperación de soporte sólo se puede realizar en el reinicio del gestor de colas y es probable que tarde una gran cantidad de tiempo, especialmente si se deben leer los registros de archivado incluidos



en cinta. Sin embargo, normalmente en esta circunstancia, el conjunto de páginas ha estado fuera de línea para el periodo de intervención y el registro no contiene ninguna información pertinente para la recuperación del conjunto de páginas.

Están disponibles las tres opciones siguientes:

#### **Permitir que se realice una recuperación de soporte completa.**

1. Detenga el gestor de colas.
2. Asegúrese de que las definiciones están disponibles para el conjunto de páginas en el procedimiento de tarea iniciada para el gestor de colas y en el conjunto de datos de inicialización de CSQINP1.
3. Reinicie el gestor de colas.

#### **Permitir la destrucción de los mensajes del conjunto de páginas.**

Esta opción es útil cuando un conjunto de páginas ha estado fuera de línea durante mucho tiempo (algunos meses, por ejemplo) y ahora se ha decidido reutilizarlo para una finalidad distinta.

1. Dé formato al conjunto de páginas utilizando la función FORMAT de CSQUTIL con la opción TYPE(NUEVO).
2. Añada definiciones para el conjunto de páginas en el procedimiento de la tarea iniciada para el gestor de colas y en el conjunto de datos de inicialización de CSQINP1.
3. Reinicie el gestor de colas.

El uso de la opción TYPE(NUEVO) para formatear borra el contenido actual del conjunto de páginas e indica al gestor de colas que ignore la información histórica en el punto de comprobación sobre el conjunto de páginas.

#### **Activar el conjunto de páginas evitando el proceso de recuperación de soporte.**

Utilice esta técnica sólo si está seguro de que el conjunto de páginas se ha desactivado desde un cierre limpio del gestor de colas. Esta opción es más apropiada cuando el conjunto de páginas se ha desactivado durante un breve periodo de tiempo, normalmente debido a problemas operativos como, por ejemplo, la ejecución de una copia de seguridad mientras se está iniciando el gestor de colas.

1. Formatee el conjunto de páginas utilizando la función FORMAT de CSQUTIL con la opción TYPE(SUSTITUIR).
2. Vuelva a añadir el conjunto de páginas en el gestor de colas dinámicamente mediante el mandato DEFINE PSID con la opción DSN o permita que se añada durante el reinicio del gestor de colas.

El uso de la opción TYPE(SUSTITUIR) para el formato comprueba que el gestor de colas cerró limpiamente el conjunto de páginas y lo marca de forma que no se vaya a realizar la recuperación de soporte. No se realiza ningún otro cambio en el contenido del conjunto de páginas.

## **Cómo hacer una copia de seguridad y recuperar conjuntos de páginas**

Existen distintos mecanismos disponibles para la copia de seguridad y la recuperación. Utilice este tema para comprender estos mecanismos.

Esta sección describe los siguientes temas:

- [“Creación de un punto de recuperación para los recursos no compartidos” en la página 505](#)
- [“Copia de seguridad de conjuntos de páginas” en la página 506](#)
- [“Recuperación de conjuntos de páginas” en la página 507](#)
- [Cómo suprimir conjuntos de páginas](#)

Para obtener información sobre cómo crear un punto de recuperación para recursos compartidos, consulte [“Recuperar colas compartidas” en la página 514](#).



## Creación de un punto de recuperación para los recursos no compartidos

IBM MQ puede recuperar objetos y mensajes persistentes no compartidos a su estado actual si se dan ambas situaciones:

1. Existen copias de conjuntos de páginas de un punto anterior.
2. Todos los registros de IBM MQ están disponibles para realizar la recuperación a partir de dicho punto.

Estos representan un punto de recuperación para los recursos no compartidos.

Tanto los objetos como los mensajes se conservan en conjuntos de páginas. Pueden existir varios objetos y mensajes de distintas colas en el mismo conjunto de páginas. Con fines de recuperación, no se pueden hacer copias de seguridad de los objetos y mensajes en aislamiento, así que se debe hacer una copia de seguridad de un conjunto de páginas como un todo para garantizar la recuperación correcta de los datos.

El registro de recuperación de IBM MQ contiene un registro de todos los mensajes persistentes y los cambios realizados en los objetos. Si IBM MQ falla (por ejemplo, debido a un error de E/S en un conjunto de páginas), puede recuperar el conjunto de páginas restaurando la copia de seguridad y reiniciando el gestor de colas. IBM MQ aplica los cambios de registro en el conjunto de páginas desde el punto de la copia de seguridad.

Existen dos formas de crear un punto de recuperación:

### Copia de seguridad completa

Detenga el gestor de colas, que obliga todas las actualizaciones en los conjuntos de páginas.

Esto le permite reiniciar desde el punto de recuperación, utilizando sólo los conjuntos de datos del conjunto de páginas y los registros a partir de dicho punto.

### Copia de seguridad difusa

Realice copias de seguridad *difusas* de los conjuntos de páginas sin detener el gestor de colas.

Si utiliza este método y los registros asociados se dañan o pierden posteriormente, no puede utilizar las copias de seguridad difusas del conjunto de páginas para recuperar. Esto se debe a que las copias de seguridad difusas del conjunto de páginas contienen una vista incoherente del estado del gestor de colas y dependen de los registros que están disponibles. Si los registros no están disponibles, deberá volver al último conjunto de copias de conjunto de páginas de copia de seguridad obtenidas mientras el subsistema estaba inactivo (Método 1) y acepte la pérdida de datos a partir de ese momento.

### Método 1: copia de seguridad completa

Este método implica concluir el gestor de colas. Esto fuerza todas las actualizaciones en los conjuntos de páginas de forma que los conjuntos de páginas están en un estado coherente.

1. Detenga todas las aplicaciones de IBM MQ que utilizan el gestor de colas (lo que les permite completarse en primer lugar). Esto se puede realizar cambiando los valores de cola o de seguridad del acceso, por ejemplo.
2. Cuando se haya completado toda la actividad, muestre y resuelva las unidades pendientes de recuperación. (Utilice los mandatos `DISPLAY CONN` y `RESOLVE INDOUBT`, tal como se describe en [DISPLAY CONN](#) y [RESOLVE INDOUBT](#).)

Esto lleva a los conjuntos de páginas a un estado coherente; si no hace esto, los conjuntos de páginas podrían no ser coherentes y estará realizando de forma efectiva una copia de seguridad difusa.

3. Emita el mandato `ARCHIVE LOG` para asegurarse de que los últimos datos de registro se escriban en los conjuntos de datos de registro.
4. Emita el mandato `STOP QMGR MODE(QUIESCE)`. Anote el valor de RBA inferior en los mensajes `CSQI024I` o `CSQI025I` (consulte [CSQI024I](#) y [CSQI025I](#) para obtener más información). Debe mantener los conjuntos de datos de registro a partir del indicado por el valor de RBA hasta el conjunto de datos de registro actual.
5. Realice copias de seguridad de todos los conjuntos de páginas del gestor de colas (consulte [“Copia de seguridad de conjuntos de páginas”](#) en la página 506).

## Método 2: copia de seguridad difusa

Este método no implica concluir el gestor de colas. Por lo tanto, las actualizaciones podrían estar en almacenamientos intermedios virtuales durante el proceso de copia de seguridad. Esto significa que los conjuntos de páginas no están en un estado coherente y sólo se pueden utilizar para la recuperación con los registros.

1. Emita el mandato DISPLAY USAGE TYPE(ALL) y anote el valor de RBA en los mensajes CSQI024I o CSQI025I (consulte [CSQI024I](#) y [CSQI025I](#) para obtener más información).
2. Realice copias de seguridad de todos los conjuntos de páginas (consulte [“Copia de seguridad de conjuntos de páginas”](#) en la página 506).
3. Emita el mandato ARCHIVE LOG, para asegurarse de que los últimos datos de registro se escriben en los conjuntos de datos de registro. Para reiniciar desde el punto de recuperación, debe mantener los conjuntos de datos de registro a partir del conjunto de datos de registro indicado por el valor de RBA hasta el conjunto de datos de registro actual.

## Copia de seguridad de conjuntos de páginas

Para recuperar un conjunto de páginas, IBM MQ debe saber hasta que punto del registro debe remontarse. IBM MQ mantiene un número RBA de registro en la página cero de cada conjunto de páginas, denominado *número de secuencia de registro de recuperación* (LSN). Este número es el RBA inicial del registro a partir del cual IBM MQ puede recuperar el conjunto de páginas. Cuando se hace una copia de seguridad de un conjunto de páginas, este número también se copia.

Si la copia se utiliza más tarde para recuperar el conjunto de páginas, IBM MQ debe tener acceso a todos los registros de anotaciones desde este valor de RBA hasta el RBA actual. Esto significa que debe mantener el número suficiente de registros de anotaciones para que IBM MQ pueda realizar la recuperación a partir de la copia de seguridad más antigua de un conjunto de páginas que tenga intención de conservar.

Utilice función ADRDSSU COPY para copiar los conjuntos de páginas.

Para obtener más información, consulte la documentación de [Sintaxis del mandato COPY DATASET para el conjunto de datos lógicos](#).

Por ejemplo:

```
//STEP2 EXEC PGM=ADRDSSU,REGION=6M
//SYSPRINT DD SYSOUT=H
//SYSIN DD *
COPY -
DATASET(INCLUDE(SCENDATA.MQPA.PAGESET.*)) -
RENAMEU(SCENDATA.MQPA.PAGESET.** , SCENDATA.MQPA.BACKUP1.** ) -
SPHERE -
REPUNC -
FASTREPLICATION(PREF ) -
CANCELERROR -
TOL(ENQF)
/*
//
```

Si copia el conjunto de páginas mientras el gestor de colas se está ejecutando, debe utilizar un programa de utilidad de copia que, en primer lugar, copia la página cero del conjunto de páginas. Si no hace esto, se podrían dañar los datos del conjunto de páginas.

Si el proceso de ampliar dinámicamente un conjunto de páginas se interrumpe, por ejemplo se activa el sistema que se está perdiendo, puede seguir utilizando ADRDSSU para realizar una copia de seguridad de un conjunto de páginas.

Si realiza un IDCAMS de servicios de método de acceso LISTCAT ENT('page set data set name') ALLOC, verá que el HI-ALLOC-RBA es superior al HI-USED-RBA.

La próxima vez que se llena este conjunto de páginas, se vuelve a ampliar, si es posible, y se utilizan las páginas entre el RBA más utilizado y el RBA más asignado, junto con otra nueva extensión.

## Copia de seguridad de las definiciones de objetos

También debe hacer copias de seguridad de las definiciones de objeto. Para ello, utilice la característica MAKEDEF de la función COMMAND de CSQUTIL (que se describe en [Emisión de mandatos a IBM MQ \(COMMAND\)](#)).

Haga una copia de seguridad de las definiciones de objeto siempre que realice una copia de seguridad del gestor de colas y conserve la versión más actual.

## Recuperación de conjuntos de páginas

Si el gestor de colas ha terminado debido a una anomalía, el gestor de colas puede reiniciarse normalmente con toda la recuperación que se realiza durante el reinicio. Sin embargo, dicha recuperación no es posible si alguno de los conjuntos de páginas o conjuntos de datos de registro no está disponible. El alcance hasta el que puede recuperar depende de la disponibilidad de las copias de seguridad de los conjuntos de páginas y los conjuntos de datos de registro.

Para reiniciar desde un punto de recuperación debe tener:

- Una copia de seguridad del conjunto de páginas que se va a recuperar.
- Si ha utilizado el proceso de copia de seguridad "difusa" descrito en ["Método 2: copia de seguridad difusa"](#) en la página 506, el conjunto de datos de registro que incluía el valor RBA anotado, el conjunto de datos de registro que fue realizado por el mandato ARCHIVE LOG y todos los conjuntos de datos de registro entre estos.
- Si ha utilizado la copia de seguridad completa, pero no tiene los conjuntos de datos de registro siguientes que fueron realizados por el mandato ARCHIVE LOG, **no** es necesario que ejecute la función FORMAT TYPE(REPLACE) del programa de utilidad CSQUTIL en todos los conjuntos de páginas.

Para recuperar un conjunto de páginas a su estado actual, también debe tener todos los conjuntos de datos de registros y todos los registros desde el mandato ARCHIVE LOG.

Existen dos métodos para recuperar un conjunto de páginas. Para utilizar cualquiera de los dos métodos, el gestor de colas debe estar detenido.

### Recuperación sencilla

Se trata del método más sencillo y es apropiado para la mayoría de las situaciones de recuperación.

1. Suprima el conjunto de páginas que desee recuperar a partir de la copia de seguridad.
2. Utilice la función ADRDSSU COPY para recuperar el conjunto de páginas de la copia de seguridad.

De forma alternativa, puede cambiar el nombre de la copia de seguridad por el nombre original, o cambiar la sentencia CSQP00xx DD en el procedimiento del gestor de colas para apuntar al conjunto de páginas de copia de seguridad. Sin embargo, si pierde o daña el conjunto de páginas, dejará de tener una copia de seguridad a partir de la cual realizar la restauración.

3. Reinicie el gestor de colas.
4. Cuando el gestor de colas se haya reiniciado correctamente, podrá reiniciar las aplicaciones
5. Restablezca los procedimientos normales de copia de seguridad para la página restaurada.

### Recuperación avanzada

Este método proporciona ventajas de rendimiento si tiene un conjunto de páginas grande para recuperar, o si ha habido mucha actividad en el conjunto de páginas desde que se realizó la última copia de seguridad. Sin embargo, requiere más intervención manual que el método sencillo, que podría aumentar el riesgo de error y el tiempo necesario para realizar la recuperación.

1. Suprima y redefina el conjunto de páginas que desee recuperar a partir de la copia de seguridad.
2. Utilice ADRDSSU para copiar la copia de seguridad del conjunto de páginas en el nuevo conjunto de páginas. Defina el nuevo conjunto de páginas con un valor de extensión secundario para que se pueda ampliar dinámicamente.

De forma alternativa, puede cambiar el nombre de la copia de seguridad por el nombre original, o cambiar la sentencia CSQP00xx DD en el procedimiento del gestor de colas para apuntar al conjunto de páginas de copia de seguridad. Sin embargo, si pierde o daña el conjunto de páginas, dejará de tener una copia de seguridad a partir de la cual realizar la restauración.

3. Cambie las definiciones CSQINP1 para el gestor de colas para que la agrupación de almacenamiento intermedio asociada con el conjunto de páginas se recupere de la forma más grande posible. Al agrandar la agrupación de almacenamiento intermedio, es posible que pueda mantener todas las páginas modificadas residentes en la agrupación de almacenamiento intermedio y reducir la cantidad de E/S en el conjunto de páginas.
4. Reinicie el gestor de colas.
5. Cuando el gestor de colas se haya reiniciado correctamente, deténgalo (utilizando la desactivación temporal) y, a continuación, reinicielo utilizando la definición de agrupación de almacenamiento intermedio para dicho conjunto de páginas. Después de que se complete correctamente este segundo reinicio, podrá reiniciar las aplicaciones
6. Restablezca los procedimientos normales de copia de seguridad para la página restaurada.

### ¿Qué sucede cuando se reinicia el gestor de colas?

Cuando el gestor de colas se reinicia, aplica todos los cambios realizados en el conjunto de páginas que están registrados en el registro, empezando desde el punto de reinicio para el conjunto de páginas. IBM MQ puede recuperar varios conjuntos de páginas de este modo. El conjunto de páginas se amplía dinámicamente, si es necesario, durante la recuperación de soporte.

Durante el reinicio, IBM MQ determina el RBA del registro desde el cual empezar a tomar el valor más bajo entre los siguientes:

- El LSN de recuperación desde el registro de anotaciones del punto de comprobación para cada conjunto de páginas.
- El LSN de recuperación desde la página cero de cada conjunto de páginas.
- El RBA de la unidad de recuperación incompleta más antigua en el sistema en el momento en que se realizó la copia de seguridad.

Todas las definiciones de objeto se almacenan en el conjunto de páginas cero. Los mensajes se pueden almacenar en cualquier conjunto de páginas disponible.

**Nota:** El gestor de colas no se puede reiniciar si el conjunto de páginas no está disponible.

### Cómo suprimir conjuntos de páginas

Puede suprimir un conjunto de páginas mediante el mandato DELETE PSID; consulte [DELETE PSID](#) para obtener más detalles de este mandato.

No puede suprimir un conjunto de páginas al que siga haciendo referencia cualquier clase de almacenamiento. Utilice DISPLAY STGCLASS para descubrir qué clases de almacenamiento hacen referencia a un conjunto de páginas.

El conjunto de datos se desasigna de IBM MQ, pero no se suprime. Permanece disponible para usos futuros, o se puede suprimir utilizando los recursos de z/OS.

Elimine el conjunto de páginas desde el procedimiento de tarea reiniciado para el gestor de colas.

Elimine la definición del conjunto de páginas del conjunto de datos de inicialización CSQINP1.

### **Cómo hacer una copia de seguridad y restaurar las colas utilizando CSQUTIL**

Utilice este tema como referencia para información adicional sobre la copia de seguridad y la restauración utilizando CSQUTIL.

Puede utilizar las funciones del programa de utilidad CSQUTIL para hacer copias de seguridad y restaurar colas. Para hacer una copia de seguridad de la cola, utilice la función COPY o SCOPY para copiar los

mensajes de una cola en un conjunto de datos. Para restaurar la cola, utilice la función complementaria LOAD o SLOAD. Para obtener más información, consulte [Programa de utilidad de IBM MQ](#).

## **Gestión de agrupaciones de almacenamientos intermedios**

Utilice este tema si desea cambiar o suprimir las agrupaciones de almacenamiento intermedio.

En este tema se describe cómo modificar y suprimir agrupaciones de almacenamiento intermedio. Contiene las siguientes secciones:

- “[Cómo cambiar el número de almacenamientos intermedios en una agrupación de almacenamiento intermedio](#)” en la página 509
- “[Cómo suprimir una agrupación de almacenamiento intermedio](#)” en la página 510

Las agrupaciones de almacenamiento intermedio se definen durante la inicialización del gestor de colas, utilizando mandatos `DEFINE BUFFPOOL` emitidos desde el conjunto de datos de entrada de inicialización CSQINP1. Se pueden modificar sus atributos de acuerdo con los requisitos empresariales mientras el gestor de colas está en ejecución, realizando los procesos que se describen en este tema. El gestor de colas registra los atributos de la agrupación de almacenamiento intermedio actual en los registros de anotaciones de punto de comprobación. Éstos se restauran automáticamente durante el siguiente reinicio del gestor de colas, a menos que la definición de agrupación de almacenamiento intermedio en CSQINP1 incluya el atributo REPLACE.

Utilice el mandato `DISPLAY USAGE` para visualizar los atributos del almacenamiento intermedio actual.

También puede definir agrupaciones de almacenamiento intermedio dinámicamente mediante el mandato `DEFINE PSID` con la opción DSN.

Si cambia agrupaciones de almacenamiento intermedio dinámicamente, también debe actualizar sus definiciones en el conjunto de datos de inicialización CSQINP1.

Consulte [Planificación en z/OS](#) para ver una descripción de conjuntos de páginas, clases de almacenamiento, almacenamientos intermedios y agrupaciones de almacenamiento intermedio, y algunas de las consideraciones de rendimiento que se aplican.

**Nota:** Las agrupaciones de almacenamientos intermedios utilizan mucho espacio de almacenamiento. Al incrementar el tamaño de una agrupación de almacenamiento intermedio o definir una nueva agrupación de almacenamiento intermedio, asegúrese de que haya suficiente almacenamiento disponible. Para obtener más información, consulte [Almacenamiento de espacios de direcciones](#).

### **Cómo cambiar el número de almacenamientos intermedios en una agrupación de almacenamiento intermedio**

Si una agrupación de almacenamiento intermedio es demasiado pequeña, la condición puede dar como resultado que se muestre el mensaje CSQP020E en la consola; puede asignarle más almacenamientos intermedios mediante el mandato `ALTER BUFFPOOL` de la siguiente manera:

1. Determine cuánto espacio hay disponible para nuevos almacenamientos intermedios consultando los mensajes `CSQY220I` en el registro. El espacio disponible se indica en MB. Como un almacenamiento intermedio tiene un tamaño de 4 KB, cada MB de espacio disponible permite asignar 256 almacenamientos intermedios. No asigne todo el espacio libre a los almacenamientos intermedios, ya que se necesita algo de espacio para otras tareas.

Si la agrupación de almacenamiento intermedio utiliza páginas fijas de 4 KB, esto es, su atributo PAGECLAS es FIXED4KB, asegúrese de que existe suficiente almacenamiento real disponible en LPAR.

2. Si el espacio libre indicado es insuficiente, libere algunos almacenamientos intermedios de otra agrupación de almacenamiento intermedio mediante el mandato

```
ALTER BUFFPOOL(buf-pool-id) BUFFERS(integer)
```

donde *id-agrup-alm-int* es la agrupación de almacenamientos intermedios de la que desea recuperar espacio y *entero* es el nuevo número de almacenamientos intermedios que se asignarán a esta agrupación de almacenamientos intermedios, que debe ser menor que el número original de almacenamientos intermedios asignados a la misma.

3. Añada almacenamientos intermedios a la agrupación de almacenamiento intermedio que desea expandir mediante el mandato

```
ALTER BUFFPOOL(buf-pool-id) BUFFERS(integer)
```

donde *id-agrup-alm-int* es la agrupación de almacenamientos intermedios que se va a expandir y *entero* es el número de almacenamientos intermedios que se asignarán a esta agrupación de almacenamientos intermedios, que debe ser mayor que el número original de almacenamientos intermedios asignados a la misma.

## Cómo suprimir una agrupación de almacenamiento intermedio

Cuando una agrupación de almacenamiento intermedio ya no sea utilizada por ningún conjunto de páginas, suprímala para liberar el almacenamiento virtual que tiene asignado.

Una agrupación de almacenamiento intermedio se suprime con el mandato `DELETE BUFFPOOL`. El mandato no se ejecuta correctamente si algún conjunto de páginas está utilizando esta agrupación de almacenamiento intermedio.

Consulte [“Cómo suprimir conjuntos de páginas”](#) en la [página 508](#) para obtener información sobre cómo suprimir los conjuntos de páginas.

## **Gestión de grupos de compartición de colas y colas compartidas en z/OS**

IBM MQ puede utilizar diferentes tipos de recursos compartidos; por ejemplo, grupos de compartición de colas, colas compartidas y el recurso de acoplamiento. Utilice este tema para revisar los procedimientos necesarios para gestionar estos recursos compartidos.

Este apartado contiene información sobre los siguientes temas:

- [“Gestión de grupos de compartición de colas”](#) en la [página 510](#)
- [“Gestión de colas compartidas”](#) en la [página 513](#)
- [“Gestión de objetos de grupo”](#) en la [página 518](#)
- [“Gestión del recurso de acoplamiento”](#) en la [página 519](#)

## **Gestión de grupos de compartición de colas**

Puede añadir o eliminar un gestor de colas en un grupo de compartición de colas (QSG) y gestionar las tablas de Db2 asociadas.

Este tema tiene secciones sobre las siguientes tareas:

- [“Configuración de un grupo de compartición de colas”](#) en la [página 510](#)
- [“Adición de un gestor de colas a un grupo de compartición de colas”](#) en la [página 511](#)
- [“Eliminación de un gestor de colas de un grupo de compartición de colas”](#) en la [página 512](#)
- [“Eliminación de un grupo de compartición de colas de las tablas de Db2”](#) en la [página 513](#)
- [“Validación de la coherencia de definiciones Db2”](#) en la [página 513](#)

## **Configuración de un grupo de compartición de colas**

Cada grupo de compartición de colas tiene un nombre de hasta cuatro caracteres. El nombre debe ser exclusivo en la red y debe ser diferente de los nombres de gestor de colas.

Siga estos pasos para configurar un grupo de compartición de colas:

1. Si este es el primer grupo de compartición de colas para utilizar el grupo de uso compartido de datos de Db2, [configure el entorno de Db2](#).
2. [Configure el recurso de acoplamiento](#).
3. Añada el grupo de compartición de colas a las tablas de Db2. Utilice la función ADD QSG del programa de utilidad del grupo de compartición de colas (CSQ5PQSG). Este programa se describe en [El programa de utilidad de grupo de compartición de colas](#). Se proporciona un ejemplo en `thlqual.SCSQPROC(CSQ45AQS)`.
4. Añada un gestor de colas al grupo de compartición de colas siguiendo los pasos que se indican en [“Adición de un gestor de colas a un grupo de compartición de colas”](#) en la página 511
5. Defina estructuras de aplicación en IBM MQ siguiendo los pasos de [“Añadir una estructura de recurso de acoplamiento”](#) en la página 519.
6. Si es necesario, [migre colas no compartidas a colas compartidas](#).
7. Para obtener disponibilidad, cree canales compartidos dentro y fuera del grupo de compartición de colas.
  - Para conexiones en el grupo de compartición de colas:
    - Configure un socket VIPA o un direccionador de hardware para distribuir la carga de trabajo entre los gestores de colas disponibles en el QSG.
    - Defina un canal receptor con QSGDISP(GROUP), para asegurarse de que la definición de canal esté disponible en todos los gestores de colas en el QSG.
    - Inicie un escucha con INDISP(GROUP), en cada gestor de colas, para las conexiones de canal MCA en el QSG. Las conexiones de cliente en el QSG deben seguir conectándose a un escucha iniciado con INDISP(QMGR).
    - Cambie las aplicaciones para conectarse utilizando el nombre QSG, en lugar de un nombre de gestor de colas específico.
    - Asegúrese de que las reglas de autenticación de canal en todos los gestores de colas en el QSG son las mismas, para permitir a las aplicaciones conectarse a cualquier gestor de colas en el QSG.
  - Para las conexiones fuera del grupo de compartición de colas:
    - Defina una cola de transmisión compartida.
    - Defina el canal de salida con QSGDISP(GROUP) y DEFCDISP(SHARED).

Si convierte un canal existente a un canal compartido, es posible que tenga que emitir el mandato [RESET CHANNEL](#) antes de iniciar el canal, ya que la cola de sincronización utilizada por el canal habrá cambiado.

## Adición de un gestor de colas a un grupo de compartición de colas

Un gestor de colas se puede añadir a un grupo de compartición de colas existente.

Tenga en cuenta que:

- El grupo de compartición de colas debe existir antes de poder añadirle gestores de colas.
- Un gestor de colas puede ser miembro de un solo grupo de compartición de colas.

Siga estos pasos para añadir un gestor de colas a un grupo de compartición de colas:

1. Realice las tareas en [la implementación de controles de seguridad de ESM para el grupo de compartición de colas](#) para otorgar el acceso apropiado a los ID de usuario del gestor de colas y del iniciador de canal.
2. Si el grupo de compartición de colas tiene estructuras CF configuradas para descargar datos en SMDS, realice las tareas en [la configuración del entorno SMDS](#).
3. Detenga el gestor de colas.



4. Utilice la función ADD QMGR del programa de utilidad del grupo de compartición de colas (CSQ5PQSG). Este programa se describe en el [programa de utilidad de grupo de compartición de colas](#). Se proporciona un ejemplo en thlqual.SCSQPROC(CSQ45AQM).
5. [Cambie el módulo del parámetro del sistema](#) para añadir datos de grupo de compartición de colas:
  - a. Modifique CSQ6SYSP para especificar el parámetro QSGDATA. Consulte la [utilización de CSQ6SYSP](#) si desea más información.
  - b. Ensamble y enlace el módulo de parámetros del sistema. Es posible que desee utilizar un nombre distinto para el módulo de carga.
  - c. Cambie el proceso de inicio para utilizar el nuevo módulo.
6. Copie y adapte el miembro de ejemplo thlqual.SCSQPROC(CSQ4INSS), que define las estructuras CF y las colas SYSTEM necesarias. Añada el miembro personalizado a CSQINP2 DD en el JCL de inicio del gestor de colas.
7. Reinicie el gestor de colas utilizando el módulo del parámetro del sistema del grupo de compartición de colas.
8. Si lo desea, migre a los perfiles de seguridad con el nombre de grupo de compartición de colas como prefijo, en lugar del nombre del gestor de colas.
9. Si los canales compartidos se utilizan para conexiones en el QSG, cree reglas de autenticación de canal que las duplique en los otros gestores de colas en el QSG, para permitir a las aplicaciones conectarse a cualquier gestor de colas en el QSG.
10. 10. Opcionalmente, realice una de las siguientes acciones para permitir que las aplicaciones conectadas al gestor de colas del QSG pongan mensajes en las colas alojadas por otros gestores de colas en el QSG:
  - Active [transferencia a colas entre grupos](#) emitiendo el mandato ALTER QMGR IGQ (ENABLED).
  - Defina colas de transmisión y canales en los otros gestores de colas en el QSG. La definición de colas de transmisión con el mismo nombre que los gestores de colas de destino evita la necesidad de definir colas remotas y alias de gestor de colas.

**Nota:** Para añadir un gestor de colas a un grupo de compartición de colas existente que contenga gestores de colas que ejecuten versiones anteriores de IBM MQ, primero debe aplicar el arreglo temporal de programa de coexistencia para la versión superior de IBM MQ del grupo a cada gestor de colas de versión anterior del grupo.

## Eliminación de un gestor de colas de un grupo de compartición de colas

Solo puede eliminar un gestor de colas de un grupo de compartición de colas si los registros del gestor de colas no son necesarios para otro proceso y todos los SMDS propiedad del gestor de colas están vacíos.

Los registros son necesarios si contienen:

- La última copia de seguridad de una de las estructuras de aplicación del recurso de acoplamiento (CF) utilizadas por el grupo de compartición de colas.
- Los datos necesarios para un futuro proceso de restauración, es decir, el gestor de colas ha utilizado una estructura recuperable desde el momento descrito por el último valor de intervalo de exclusión de copia de seguridad.

Si se aplica alguno de estos puntos, o ambos, o un SMDS propiedad del gestor de colas contiene mensajes, el gestor de colas no se puede eliminar. Para determina qué registros de los gestores de colas son necesarios para un futuro proceso de restauración, utilice el mandato MQSC DISPLAY CFSTATUS con la opción TYPE(BACKUP) (para obtener información detallada sobre este mandato, consulte [DISPLAY CFSTATUS](#)).

Utilice los pasos siguientes para eliminar un gestor de colas de un grupo de compartición de colas:

1. Detenga cualquier aplicación conectada al gestor de colas que coloca mensajes en colas compartidas.



2. Resuelva las unidades dudosas de trabajo que implican este gestor de colas.
3. Determine si existe algún mensaje en cualquier SMDS propiedad del gestor de colas emitiendo el mandato DISPLAY USAGE TYPE(SMDS).
4. Si hay mensajes descargados para cualquier estructura de aplicación, espere hasta que estos mensajes se hayan recuperado de la cola. El número de mensajes descargados notificado por DISPLAY USAGE TYPE(SMDS) debería ser cero antes de continuar.
5. Concluya el gestor de colas utilizando STOP QMGR MODE(QUIESCE).
6. Espere un intervalo, como mínimo, equivalente al valor del parámetro EXCLINT que especificará en el mandato BACKUP CFSTRUCT en el paso siguiente.
7. En otro gestor de colas, ejecute una copia de seguridad de estructura CF para cada estructura CF recuperable mediante el mandato MQSC BACKUP CFSTRUCT y especificando un valor EXCLINT según sea necesario en el paso anterior.
8. Confirme que los registros del gestor de colas no son necesarios para restaurar ninguna estructura CF, inspeccionando la salida del mandato DISPLAY CFSTATUS (\*) TYPE (BACKUP).
9. Utilice la función REMOVE QMGR del programa de utilidad CSQ5PQSG para eliminar el gestor de colas del grupo de compartición de colas. Este programa se describe en [El programa de utilidad de grupo de compartición de colas](#). Se proporciona un ejemplo en thlqual.SCSQPROC(CSQ45RQM).
10. Antes de reiniciar el gestor de colas, restablezca el parámetro de sistema QSGDATA a su valor predeterminado, y vuelva a crear el módulo de parámetro de sistema. Consulte [Utilización de CSQ6SYSP](#) para obtener información sobre cómo adaptar los requisitos del sistema.

Observe que, al eliminar el último gestor de colas de un grupo de compartición de colas, debe utilizar la opción FORCE en lugar de REMOVE. Esto elimina el gestor de colas del grupo de compartición de colas, sin realizar las comprobaciones de coherencia de los registros de gestor de colas que se necesitan para la recuperación. Realice esta operación sólo si va a suprimir el grupo de compartición de colas.

## Eliminación de un grupo de compartición de colas de las tablas de Db2

Para eliminar un grupo de compartición de colas de las tablas Db2, utilice la función REMOVE QSG del programa de utilidad del grupo de compartición de colas (CSQ5PQSG). Este programa se describe en [El programa de utilidad de grupo de compartición de colas](#). Se proporciona un ejemplo en thlqual.SCSQPROC(CSQ45RQS).

Solo puede eliminar un grupo de compartición de colas de las tablas comunes del grupo de uso compartido de datos de Db2 después de haber eliminado todos los gestores de colas del grupo de compartición de colas (tal como se describe en [“Eliminación de un gestor de colas de un grupo de compartición de colas”](#) en la página 512).

Cuando el registro del grupo de compartición de colas se suprime de la tabla de administración de grupos de compartición de colas, todos los objetos y la información administrativa relacionada con dicho grupo de compartición de colas se suprimen de otras tablas de IBM MQ Db2. Así se incluye información de cola compartida y de objeto de grupo.

## Validación de la coherencia de definiciones Db2

Se pueden producir problemas para las colas compartidas dentro de un grupo de compartición de colas si las definiciones de objeto de Db2 se vuelven incoherentes por algún motivo.

Para validar la coherencia de las definiciones de objeto de Db2 para los gestores de colas, las estructuras de recurso de acoplamiento y las colas compartidas, utilice la función VERIFY QSG del programa de utilidad de grupo de compartición de colas (CSQ5PQSG). Este programa se describe en [El programa de utilidad de grupo de compartición de colas](#).

### **Gestión de colas compartidas**

Lea este tema para informarse sobre cómo recuperar, mover y migrar colas compartidas.

En este apartado se describen las tareas siguientes:

- [“Recuperar colas compartidas” en la página 514](#)
- [“Mover colas compartidas” en la página 514](#)
- [“Migrar colas no compartidas a colas compartidas” en la página 517](#)
- [Suspender una conexión de Db2](#)

## Recuperar colas compartidas

IBM MQ puede recuperar mensajes persistentes en colas compartidas si:

- Se han realizado copias de seguridad de las estructuras CF que contienen los mensajes.
- Están disponibles todos los registros para todos los gestores de colas del grupo de compartición de colas, para realizar la recuperación desde el punto en que se han realizado las copias de seguridad.
- Db2 está disponible y la tabla de copias de seguridad de estructura es más reciente que la copia de seguridad de estructura CF más reciente.

Los mensajes de una cola compartida se almacenan en una estructura del recurso de acoplamiento (CF). Los mensajes persistentes se pueden colocar en colas compartidas, y como los mensajes persistentes en colas no compartidas, se copian en el registro del gestor de colas. Los mandatos MQSC `BACKUP CFSTRUCT` y `RECOVER CFSTRUCT` se proporcionan para permitir la recuperación de una estructura CF en el improbable caso de una anomalía del recurso de acoplamiento. En tales circunstancias, los mensajes no persistentes almacenados en la estructura afectada se pierden, pero los mensajes persistentes se pueden recuperar. Se impide cualquier otra actividad de aplicaciones que utilice la estructura hasta que la estructura se haya recuperado.

Para habilitar la recuperación, debe realizar una copia de seguridad de las estructuras de lista del recurso de acoplamiento que utilizan con frecuencia el mandato MQSC `BACKUP CFSTRUCT`. Los mensajes de la estructura CF se graban en el conjunto de datos de registro activo del gestor de colas que hace la copia de seguridad. Se graba un registro de la copia de seguridad en Db2: el nombre de la estructura CF de la que se está realizando una copia de seguridad, el nombre del gestor de colas que realiza la copia de seguridad, el rango de RBA para esta copia de seguridad en el registro de ese gestor de colas y la hora de la copia de seguridad. Haga una copia de seguridad de las estructuras de lista de CF incluso si no está utilizando activamente colas compartidas, por ejemplo, si ha configurado un grupo de compartición de colas con intención de utilizarlo en el futuro.

Puede recuperar una estructura CF emitiendo un mandato MQSC `RECOVER CFSTRUCT` para el gestor de colas que puede realizar la recuperación; puede utilizar cualquier gestor de colas del grupo de compartición de colas. Puede especificar una sola estructura CF para recuperar, o puede recuperar varias estructuras CF simultáneamente.

Como se ha señalado antes, es importante que haga una copia de seguridad de las estructuras de listas de CF con frecuencia, de lo contrario la recuperación de una estructura CF puede tardar mucho tiempo. Además, el proceso de recuperación no se puede cancelar.

La definición de una cola compartida se guarda en una base de datos Db2 y, por lo tanto, se puede recuperar si es necesario utilizando los procedimientos de base de datos Db2 estándar. Consulte [Grupos de compartición de colas y colas compartidas](#) para obtener más información.

## Mover colas compartidas

En este apartado se describe cómo realizar un equilibrio de carga moviendo una cola compartida de una estructura de recurso de acoplamiento a otra. También se describe cómo mover una cola no compartida a una cola compartida, y cómo mover una cola compartida a una cola no compartida.

Cuando se mueve una cola, se tiene que definir una cola temporal como parte del procedimiento. Esto se debe a que cada cola debe tener un nombre exclusivo, por lo que no puede tener dos colas con el mismo

nombre, incluso si las colas tienen disposiciones de cola diferentes. IBM MQ tolera tener dos colas con el mismo nombre (como en el paso “2” en la página 515), pero no se pueden utilizar las colas.

- Mover una cola de una estructura de recurso de acoplamiento a otra
- Mover una cola no compartida a una cola compartida
- Mover una cola compartida a una cola no compartida

### Mover una cola de una estructura de recurso de acoplamiento a otra

Para mover colas y sus mensajes de una estructura CF a otra, utilice el mandato `MQSC MOVE QLOCAL`. Cuando haya identificado la cola o colas que desea mover a una nueva estructura CF, utilice el procedimiento siguiente para mover cada cola:

1. Asegúrese de que la cola que desea mover no está siendo utilizada por ninguna aplicación, es decir, los atributos de cola `IPPROCS` y `OPPROCS` tienen un valor de cero en todos los gestores de colas del grupo de compartición de colas.
2. Impida que las aplicaciones pongan mensajes en la cola que se está moviendo alterando la definición de cola para inhabilitar `MQPUT`. Cambie la definición de cola por `PUT (DISABLED)`.
3. Defina una cola temporal con los mismos atributos que la cola que se va a mover mediante el mandato siguiente :

```
DEFINE QL(TEMP_QUEUE) LIKE(QUEUE_TO_MOVE) PUT(ENABLED) GET(ENABLED) QSGDISP(QMGR)
```

**Nota:** Si esta cola temporal ya existe de una ejecución anterior, suprimala antes de realizar la definición.

4. Mueva los mensajes a la cola temporal mediante el mandato siguiente:

```
MOVE QLOCAL(QUEUE_TO_MOVE) TOQLOCAL(TEMP_QUEUE)
```

5. Suprima la cola que está moviendo, mediante el mandato:

```
DELETE QLOCAL(QUEUE_TO_MOVE)
```

6. Redefina la cola que se está moviendo, cambiando el atributo `CFSTRUCT`, con el mandato siguiente:

```
DEFINE QL(QUEUE_TO_MOVE) LIKE(TEMP_QUEUE) CFSTRUCT(NEW) QSGDISP(SHARED)
```

Cuando se redefine la cola, ésta se basa en la cola temporal que se ha creado en el paso “3” en la página 515.

7. Mueva los mensajes a la nueva cola mediante el mandato:

```
MOVE QLOCAL(TEMP) TOQLOCAL(QUEUE_TO_MOVE)
```

8. La cola creada en el paso “3” en la página 515 ya no es necesaria. Utilice el mandato siguiente para suprimirla:

```
DELETE QL(TEMP_QUEUE)
```

9. Si la cola que se está moviendo se definió en los conjuntos de datos CSQINP2, cambie el atributo CFSTRUCT del mandato DEFINE QLOCAL adecuado en los conjuntos de datos CSQINP2. Añada la palabra clave REPLACE de modo que se sustituya la definición de cola existente.

La Figura 41 en la página 516 muestra un trabajo de ejemplo para mover una cola de una estructura CF a otra.

```
//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=thlqual.SCSQANLE,DISP=SHR
//      DD DSN=thlqual.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
ALTER QL(Queue_To_Move) PUT(DISABLED)
DELETE QL(TEMP_QUEUE) PURGE
DEFINE QL(TEMP_QUEUE) LIKE(Queue_To_Move) PUT(ENABLED) GET(ENABLED) QSGDISP(QMGR)
MOVE QLOCAL(Queue_To_Move) TOQLOCAL(TEMP_QUEUE)
DELETE QL(Queue_To_Move)
DEFINE QL(Queue_To_Move) LIKE(TEMP_QUEUE) CFSTRUCT(NEW) QSGDISP(SHARED)
MOVE QLOCAL(TEMP_QUEUE) TOQLOCAL(Queue_To_Move)
DELETE QL(TEMP_QUEUE)
/*
```

Figura 41. Trabajo de ejemplo para mover una cola de una estructura CF a otra

### Mover una cola no compartida a una cola compartida

El procedimiento para mover una cola no compartida a una cola compartida es como el procedimiento para mover una cola de una estructura CF a otra (consulte “Mover una cola de una estructura de recurso de acoplamiento a otra” en la página 515). En la Figura 42 en la página 516 se ofrece un trabajo de ejemplo de este procedimiento.

**Nota:** Recuerde que los mensajes en colas compartidas están sujetos a determinadas restricciones en el tamaño máximo del mensaje, la persistencia de mensajes y el tipo de índice de cola, por lo que quizá no pueda mover algunas colas no compartidas a una cola compartida.

```
//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=thlqual.SCSQANLE,DISP=SHR
//      DD DSN=thlqual.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
ALTER QL(Queue_To_Move) PUT(DISABLED)
DELETE QL(TEMP_QUEUE) PURGE
DEFINE QL(TEMP_QUEUE) LIKE(Queue_To_Move) PUT(ENABLED) GET(ENABLED)
MOVE QLOCAL(Queue_To_Move) TOQLOCAL(TEMP_QUEUE)
DELETE QL(Queue_To_Move)
DEFINE QL(Queue_To_Move) LIKE(TEMP_QUEUE) CFSTRUCT(NEW) QSGDISP(SHARED)
MOVE QLOCAL(TEMP_QUEUE) TOQLOCAL(Queue_To_Move)
DELETE QL(TEMP_QUEUE)
/*
```

Figura 42. Trabajo de ejemplo para mover una cola no compartida a una cola compartida

## Mover una cola compartida a una cola no compartida

El procedimiento para mover una cola compartida a una cola no compartida es como el procedimiento para mover una cola de una estructura CF a otra (consulte [“Mover una cola de una estructura de recurso de acoplamiento a otra”](#) en la página 515).

En la [Figura 43 en la página 517](#) se ofrece un trabajo de ejemplo de este procedimiento.

```
//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=th1qua1.SCSQANLE,DISP=SHR
// DD DSN=th1qua1.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
ALTER QL(Queue_TO_MOVE) PUT(DISABLED)
DELETE QL(TEMP_QUEUE) PURGE
DEFINE QL(TEMP_QUEUE) LIKE(Queue_TO_MOVE) PUT(ENABLED) GET(ENABLED) QSGDISP(QMGR)
MOVE QLOCAL(Queue_TO_MOVE) TOQLOCAL(TEMP_QUEUE)
DELETE QL(Queue_TO_MOVE)
DEFINE QL(Queue_TO_MOVE) LIKE(TEMP_QUEUE) STGCLASS(NEW) QSGDISP(QMGR)
MOVE QLOCAL(TEMP_QUEUE) TOQLOCAL(Queue_TO_MOVE)
DELETE QL(TEMP_QUEUE)
/*
```

*Figura 43. Trabajo de ejemplo para mover una cola compartida a una cola no compartida*

## Migrar colas no compartidas a colas compartidas

Hay dos etapas para la migración de colas no compartidas a colas compartidas:

- Migrar el primer (o único) gestor de colas del grupo de compartición de colas
- Migrar todos los demás gestores de colas del grupo de compartición de colas

### Migrar el primer (o único) gestor de colas del grupo de compartición de colas

En la [Figura 42 en la página 516](#) se muestra un trabajo de ejemplo para mover una cola no compartida a una cola compartida. Haga esto para cada cola que haya que migrar.

#### Nota:

1. Los mensajes en colas compartidas están sujetos a determinadas restricciones en el tamaño máximo del mensaje, la persistencia de mensajes y el tipo de índice de cola, por lo que quizá no pueda mover algunas colas no compartidas a una cola compartida.
2. Debe utilizar el tipo de índice correcto para colas compartidas. Si migra una cola de transmisión para que sea una cola compartida, el tipo de índice debe ser MSGID.

Si la cola está vacía, o no necesita conservar los mensajes que hay en ella, la migración de la cola es más sencilla. La [Figura 44 en la página 518](#) muestra un trabajo de ejemplo para utilizar en estas circunstancias.

```
//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=th1qua1.SCSQANLE,DISP=SHR
// DD DSN=th1qua1.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
DELETE QL(TEMP_QUEUE) PURGE
DEFINE QL(TEMP_QUEUE) LIKE(QUEUE_TO_MOVE) PUT(ENABLED) GET(ENABLED)
DELETE QL(QUEUE_TO_MOVE)
DEFINE QL(QUEUE_TO_MOVE) LIKE(TEMP_QUEUE) CFSTRUCT(NEW) QSGDISP(SHARED)
DELETE QL(TEMP_QUEUE)
/*
```

Figura 44. Trabajo de ejemplo para mover una cola no compartida sin mensajes a una cola compartida

### Migrar todos los demás gestores de colas del grupo de compartición de colas

1. Para cada cola que no tenga el mismo nombre en una cola compartida existente, mueva la cola como se describe en la [Figura 42](#) en la página 516 o en la [Figura 44](#) en la página 518.
2. Para las colas que tengan el mismo nombre que una cola compartida existente, mueva los mensajes a la cola compartida mediante los mandatos que se muestran en la [Figura 45](#) en la página 518.

```
MOVE QLOCAL(QUEUE_TO_MOVE) QSGDISP(QMGR) TOQLOCAL(QUEUE_TO_MOVE)
DELETÉ QLOCAL(QUEUE_TO_MOVE) QSGDISP(QMGR)
```

Figura 45. Mover mensajes de una cola no compartida a una cola compartida existente

### Suspender una conexión a Db2

Si desea aplicar mantenimiento o servicio a las tablas de Db2 o el paquete relacionado con las colas compartidas sin detener el gestor de colas, debe desconectar temporalmente los gestores de colas en el grupo de compartición de datos (DSG) de Db2.

Para ello:

1. Utilice el mandato MQSC [SUSPEND QMGR FACILITY](#)( Db2 ).
2. Realice los enlaces.
3. Vuelva a conectarse a Db2 con el mandato MQSC [RESUME QMGR FACILITY](#)(Db2).

Tenga en cuenta que hay restricciones en el uso de estos mandatos.



**Atención:** Mientras que la conexión de Db2 esté suspendida, las siguientes operaciones no estarán disponibles. Por lo tanto, debe hacer este trabajo cuando la empresa tenga menos actividad.

- Acceder a objetos de cola compartida para la administración (definir, suprimir, modificar)
- Iniciar canales compartidos
- Almacenamiento de mensajes en Db2
- Realizar una copia de seguridad o recuperar CFSTRUCT

### **Gestión de objetos de grupo**

Utilice este tema para comprender cómo trabajar con objetos de grupo.

IBM MQ copia automáticamente la definición de un objeto de grupo en el conjunto de páginas cero de cada gestor de colas que lo utiliza. Puede modificar la copia de la definición temporalmente, y IBM MQ le permite renovar las copias del conjunto de páginas de la copia del repositorio. IBM MQ siempre trata de renovar las copias del conjunto de páginas de la copia del repositorio durante el inicio (para objetos de canal, esto se hace cuando se reinicia el iniciador de canal). Esto garantiza que las copias del conjunto de páginas reflejen la versión en el repositorio, incluidos todos los cambios que se hayan realizado cuando el gestor de colas estaba inactivo.

Hay situaciones en las que la renovación no se lleva a cabo, por ejemplo:

- Si una copia de la cola está abierta, una renovación que cambie el uso de la cola no se realiza correctamente.
- Si una copia de una cola contiene mensajes, una renovación que suprima esa cola no se realiza correctamente.

En estas circunstancias, la renovación no se realiza en dicha copia, pero se lleva a cabo en las copias de todos los demás gestores de colas. Compruebe y corrija cualquier problema con los objetos de copia después de añadir, cambiar o suprimir un objeto de grupo, y en el reinicio del gestor de colas o el iniciador de canal.

## **Gestión del recurso de acoplamiento**

Utilice este tema para aprender a añadir o eliminar estructuras de recurso de acoplamiento (CF).

En este apartado se describen las tareas siguientes:

- [“Añadir una estructura de recurso de acoplamiento” en la página 519](#)
- [“Eliminar una estructura de recurso de acoplamiento” en la página 519](#)

### **Añadir una estructura de recurso de acoplamiento**

Para añadir una estructura de recurso de asociación, utilice el procedimiento siguiente:

1. Defina la estructura CF en el conjunto de datos de política CFRM. La información sobre cómo configurar el recurso de acoplamiento en [Configuración del recurso de acoplamiento](#) describe las reglas para denominar las estructuras de recurso de acoplamiento y cómo definir estructuras en el conjunto de datos de política CFRM.
2. Si desea configurar la estructura para descargar datos de mensaje en SMDS, asigne conjuntos de datos y déles un formato previo. Consulte [creación de un conjunto de datos de mensajes compartido](#) para ver detalles.
3. Defina la estructura en IBM MQ utilizando el mandato [DEFINE CFSTRUCT](#).

### **Eliminar una estructura de recurso de acoplamiento**

Para eliminar una estructura de recurso de asociación, utilice el procedimiento siguiente:

1. Utilice el siguiente mandato para obtener una lista de todas las colas que utilizan la estructura de recurso de acoplamiento que desea suprimir:

```
DISPLAY QUEUE(*) QSGDISP(SHARED) CFSTRUCT(structure-name)
```

2. Suprima todas las colas que utilizan la estructura.
3. Suprima la estructura CF de IBM MQ utilizando el mandato [DELETE CFSTRUCT](#).
4. Si la estructura se ha configurado para descargar datos de mensaje en SMDS, suprima SMDS.

5. Elimine la definición de estructura del conjunto de datos de política de CFRM y ejecute el programa de utilidad IXCMIAPU. (Esto es lo contrario de la tarea de personalización de personalización del recurso de acoplamiento, tal como se describe en [Configurar el recurso de acoplamiento](#).)

## **Ajuste de la supervisión de la lista de recurso de acoplamiento**

Utilice este tema para comprender la supervisión de la lista de recursos de acoplamiento

Se utiliza la supervisión de la lista de recursos de acoplamiento (CF) para supervisar el estado de estructuras de lista que contienen colas compartidas de IBM MQ. Cuando se añade un mensaje a una cola compartida, y la profundidad de la cola oscila de cero a no cero, el recurso CF informa a todos los gestores de colas del grupo de compartición de colas. Cuando se notifica, los gestores de colas podrían realizar una serie de acciones, incluyendo notificar a supervisores desencadenantes que están utilizando TRIGGER(FIRST), o aplicaciones que realizan la acción obtener-esperar.

De forma predeterminada, CF informa a todos los gestores de colas del grupo de compartición de colas a la vez. En determinadas configuraciones, esto puede provocar problemas como por ejemplo:

- Una distribución de carga de trabajo desviada, en la que un gran porcentaje de mensajes van a un gestor de colas concreto del grupo de compartición de colas, a menudo el gestor de colas se ejecuta en la LPAR más rápida o la que está más cerca del CF, o
- Se obtiene un gran número de errores, lo que provoca muchas pérdidas de tiempo en la CPU.

z/OS V2R3 presenta un nuevo atributo de gestor de recursos del recurso de acoplamiento (CFRM) llamado **KEYRNOTIFYDELAY**, que se puede utilizar con estructuras de listas que contienen colas compartidas (es decir, estructuras de aplicación y no la estructura administrativa), y que pueden, para determinadas cargas de trabajo, minimizar los efectos del desvío de la carga de trabajo y llamadas MQGET vacías o llamadas MQGET vacías.

**KEYRNOTIFYDELAY** solo se puede establecer en estructuras en un CF, que se ejecutan en CFLEVEL 22 o superior.

Su valor debe ser un número entre uno o siete dígitos decimales, en un rango de 0 a 1.000.000 microsegundos. Si está establecido en un valor distinto a cero y la profundidad de una cola oscila de cero a no cero, el CF selecciona un solo gestor de colas en el grupo de compartición de colas e informa a dicho gestor de colas antes que a los otros gestores de colas del grupo.

El gestor de colas se selecciona de una forma circular. Si el gestor de colas seleccionado no procesa el mensaje dentro del intervalo de tiempo descrito por **KEYRNOTIFYDELAY**, también se informará a todos los demás gestores de colas del grupo de compartición de colas.

Más información sobre **KEYRNOTIFYDELAY** está disponible aquí: [Comprender el retardo de notificación de supervisión de rango de claves](#).

Tenga en cuenta que hay dos atributos CFRM similares llamados **LISTNOTIFYDELAY** y **SUBNOTIFYDELAY**. Ninguno de estos tiene ningún efecto cuantificable sobre la carga de trabajo de IBM MQ.

## **Recuperación y reinicio en z/OS**

En este tema se explican los mecanismos de recuperación y reinicio utilizados por IBM MQ.

### **reiniciarIBM MQ**

Después de que un gestor de colas termina, existen diferentes procedimientos de reinicio necesarios en función de cómo ha terminado el gestor de colas. Lea este tema para conocer los diferentes procedimientos de reinicio que puede utilizar.

Este tema contiene información sobre cómo reiniciar el gestor de colas en los casos siguientes:

- [“Reiniciar después de una conclusión normal” en la página 521](#)
- [“Reiniciar después de una terminación anómala” en la página 521](#)
- [“Reiniciar si ha perdido sus conjuntos de páginas” en la página 521](#)



- [“Reiniciar si ha perdido sus conjuntos de datos de registro” en la página 521](#)
- [Reiniciar si ha perdido sus estructuras CF](#)

## Reiniciar después de una conclusión normal

Si el gestor de colas se ha detenido con el mandato STOP QMGR, el sistema finaliza el trabajo de forma ordenada y establece un punto de comprobación de terminación antes de detenerse. Cuando se reinicia el gestor de colas, éste utiliza información del registro de recuperación y de punto de comprobación del sistema para determinar el estado del sistema en la conclusión.

Para reiniciar el gestor de colas, emita el mandato START QMGR como se describe en [“Inicio y parada de un gestor de colas en z/OS” en la página 447](#).

## Reiniciar después de una terminación anómala

IBM MQ detecta automáticamente si el reinicio sigue a una conclusión normal o a una terminación anómala.

Iniciar el gestor de colas después de que haya terminado de forma anómala es diferente de iniciarlo después de que se haya emitido el mandato STOP QMGR. Si el gestor de colas termina de forma anómala, termina sin poder finalizar su trabajo o tomar un punto de comprobación de terminación.

Para reiniciar el gestor de colas, emita el mandato START QMGR como se describe en [“Inicio y parada de un gestor de colas en z/OS” en la página 447](#). Cuando se reinicia un gestor de colas después de una terminación anómala, se renueva el conocimiento de su estado en la terminación utilizando la información del registro, y se le informa del estado de diversas tareas.

Normalmente, el proceso de reinicio resuelve todos los estados incoherentes. No obstante, en algunos casos, deberá tomar medidas concretas para resolver las incoherencias. Este tema se describe en el apartado [“Recuperar unidades de trabajo manualmente” en la página 535](#).

## Reiniciar si ha perdido sus conjuntos de páginas

Si ha perdido sus conjuntos de páginas, tiene que restaurarlos desde las copias de seguridad antes de poder reiniciar el gestor de colas. Este tema se describe en el apartado [“Cómo hacer una copia de seguridad y recuperar conjuntos de páginas” en la página 504](#).

El gestor de colas puede tardar mucho en reiniciarse en estas circunstancias debido al tiempo que se necesita para realizar la recuperación desde soporte de almacenamiento.

## Reiniciar si ha perdido sus conjuntos de datos de registro

Si, tras detener un gestor de colas (mediante el mandato STOP QMGR), ambas copias del registro se pierden o quedan dañadas, puede reiniciar el gestor de colas siempre que tenga un conjunto coherente de conjuntos de páginas (generados mediante el [Método 1: Copia completa](#)).

Siga este procedimiento:

1. Defina nuevos conjuntos de páginas que correspondan a cada conjunto de páginas existente en el gestor de colas. Consulte [Tarea 15: Definir los conjuntos de páginas](#) para obtener información sobre la definición de conjuntos de páginas.  
Asegúrese de que cada nuevo conjunto de páginas sea mayor que el conjunto de páginas de origen correspondiente.
2. Utilice la función FORMAT de CSQUTIL para dar formato al conjunto de páginas de destino. Consulte [Dar formato a conjuntos de páginas](#) para obtener más información.

3. Utilice la función RESETPAGE de CSQUTIL para copiar los conjuntos de páginas existentes o restablecerlos, y restablecer la RBA de registro en cada página. Consulte [Copiar un conjunto de páginas y restablecer el registro](#) para obtener más información sobre esta función.
4. Redefina los conjuntos de datos de registro y el BSDS del gestor de colas utilizando CSQJU003 (consulte [El programa de utilidad de inventario de registro de cambios](#)).
5. Reinicie el gestor de colas utilizando los nuevos conjuntos de páginas. Para ello, realice una de las siguientes acciones:
  - Cambie el procedimiento de tarea iniciada del gestor de colas para que haga referencia a los nuevos conjuntos de páginas. Consulte la [Tarea 6: Crear procedimientos para el gestor de colas de IBM MQ](#) para obtener más información.
  - Utilice los Servicios de método de acceso para suprimir los conjuntos de páginas antiguos y luego renombrar los nuevos conjuntos de páginas, asignándoles los mismos nombres que los conjuntos de páginas antiguos.

**Atención:** Antes de suprimir cualquier conjunto de páginas de IBM MQ, asegúrese de que ha realizado las copias de seguridad necesarias.

Si el gestor de colas es miembro de un grupo de compartición de colas, las definiciones de objeto GROUP y SHARED normalmente no se ven afectadas por los registros perdidos o dañados. Sin embargo, si algún mensaje de cola compartida forma parte de una unidad de trabajo que estaba cubierta por los registros perdidos o dañados, el efecto en dichos mensajes sin confirmar es imprevisible.

**Nota:** Si los registros están dañados y el gestor de colas es miembro de un grupo de compartición de colas, se puede perder la posibilidad de recuperar mensajes persistentes compartidos. Emita un mandato BACKUP CFSTRUCT inmediatamente en otro gestor de colas activo del grupo de compartición de colas para todas las estructuras CF con el atributo RECOVER(YES).

## Reiniciar si ha perdido sus estructuras CF

No es necesario reiniciar si pierde sus estructuras CF, ya que el gestor de colas no termina.

### **Recuperación de sitio alternativo en z/OS**

Puede recuperar un solo gestor de colas o un grupo de compartición de colas, o considerar la copia del disco.

Consulte las siguientes secciones para obtener más información:

- [Recuperación de un único gestor de colas en un sitio alternativo](#)
- [Recuperación de un grupo de compartición de colas.](#)
  - [Recuperación desde soporte de almacenamiento de estructuras CF](#)
  - [Copia de seguridad del grupo de compartición de colas en el sitio principal](#)
  - [Recuperación de un grupo de compartición de colas en el sitio alternativo](#)
- [Utilización de la duplicación de disco](#)

## Recuperación de un único gestor de colas en un sitio alternativo

Si se produce una pérdida total de un centro de cálculo de IBM MQ, puede realizar la recuperación en otro gestor de colas o en un grupo de compartición de colas en un sitio de recuperación. (Consulte [“Recuperación de un grupo de compartición de colas en el sitio alternativo”](#) en la página 526 para ver el procedimiento de recuperación de sitio alternativo para un grupo de compartición de colas).

Para realizar la recuperación en otro gestor de colas en un sitio de recuperación, debe hacer regularmente copias de seguridad de los conjuntos de páginas y los registros. Al igual que con todas las operaciones de recuperación de datos, los objetivos de la recuperación tras desastre son perder la menor cantidad de datos, proceso de carga de trabajo (actualizaciones) y tiempo posible.

En el sitio de recuperación :

- Los gestores de colas de recuperación **deben** tener los mismos nombres que los gestores de colas perdidos.
- El módulo de parámetros del sistema (por ejemplo, CSQZPARM) utilizado en cada gestor de colas de recuperación debe contener los mismos parámetros que el gestor de colas perdido correspondiente.

Una vez hecho esto, restablezca todos los gestores de colas tal como se describe en el siguiente procedimiento. Este se puede utilizar para realizar la recuperación tras desastre en el sitio de recuperación para un único gestor de colas. Se presupone que todo lo que está disponible son:

- Copias de los registros de archivado y los BSDS creadas mediante la ejecución normal en el sitio primario (los registros activos se habrán perdido junto con el gestor de colas en el sitio primario).
- Copias de los conjuntos de páginas del gestor de colas en el sitio primario que tienen la misma antigüedad o son más antiguas que las copias de registro de archivado más recientes disponibles.

Puede utilizar el registro dual para los registros activos y de archivado, en cuyo caso debe aplicar las actualizaciones de BSDS a las dos copias:

1. Defina nuevos conjuntos de datos de conjunto de páginas y cárguelos con los datos de las copias de los conjuntos de páginas del sitio primario.
2. Defina nuevos conjuntos de datos de registro activo.
3. Defina un nuevo conjunto de datos BSDS y utilice el mandato REPRO de los Servicios de método de acceso para copiar el BSDS archivado más reciente en él.
4. Utilice el programa de utilidad imprimir correlación de registros, CSQJU004, para imprimir información de este BSDS más reciente. En el momento en que se archivó este BSDS, el registro archivado más reciente disponible se hubiera truncado como un registro activo y no aparece como un registro archivado. Anote la dirección de byte relativa (RBA) inicial (STARTRBA) y la final (ENDRBA) de este registro.
5. Utilice el programa de utilidad de inventario de registro de cambios, CSQJU003, para registrar este conjunto de datos de registro de archivado más reciente en el BSDS que acaba de restaurar, utilizando la STARTRBA y ENDRBA que ha anotado en el Paso “4” en la página 523.
6. Utilice la opción DELETE de CSQJU003 para eliminar toda la información de registro activo del BSDS.
7. Utilice la opción NEWLOG de CSQJU003 para añadir registros activos al BSDS, no especifique la STARTRBA o ENDRBA.
8. Utilice CSQJU003 para añadir un registro de control de reinicio para el BSDS. Especifique CRESTART CREATE, ENDRBA=highrba, donde highrba es la RBA alta del registro de archivo más reciente disponible (que se encuentra en el Paso “4” en la página 523), más 1.

El BSDS ahora describe todos los registros activos como vacíos, todos los registros de archivados que tiene disponibles, y ningún punto de comprobación después del final de los registros.

9. Reinicie el gestor de colas con el mandato START QMGR. Durante la inicialización, se emite un mensaje de respuesta del operador como el siguiente:

```
CSQJ245D +CSQ1 RESTART CONTROL INDICATES TRUNCATION AT RBA highrba.
REPLY Y TO CONTINUE, N TO CANCEL
```

Escriba Y para iniciar el gestor de colas. El gestor de colas se inicia, y recupera datos hasta la ENDRBA especificada en la sentencia CRESTART.

Consulte [Utilización de los programas de utilidad de IBM MQ](#) para obtener información sobre el uso de CSQJU003 y CSQJU004.

El siguiente ejemplo muestra sentencias de entrada de ejemplo para CSQJU003 para los pasos 6, 7 y 8:

```
* Step 6
DELETE DSNAME=MQM2.LOGCOPY1.DS01
DELETE DSNAME=MQM2.LOGCOPY1.DS02
DELETE DSNAME=MQM2.LOGCOPY1.DS03
DELETE DSNAME=MQM2.LOGCOPY1.DS04
DELETE DSNAME=MQM2.LOGCOPY2.DS01
```

```

DELETE DSNNAME=MQM2.LOGCOPY2.DS02
DELETE DSNNAME=MQM2.LOGCOPY2.DS03
DELETE DSNNAME=MQM2.LOGCOPY2.DS04

* Step 7
NEWLOG DSNNAME=MQM2.LOGCOPY1.DS01,COPY1
NEWLOG DSNNAME=MQM2.LOGCOPY1.DS02,COPY1
NEWLOG DSNNAME=MQM2.LOGCOPY1.DS03,COPY1
NEWLOG DSNNAME=MQM2.LOGCOPY1.DS04,COPY1
NEWLOG DSNNAME=MQM2.LOGCOPY2.DS01,COPY2
NEWLOG DSNNAME=MQM2.LOGCOPY2.DS02,COPY2
NEWLOG DSNNAME=MQM2.LOGCOPY2.DS03,COPY2
NEWLOG DSNNAME=MQM2.LOGCOPY2.DS04,COPY2

* Step 8
CRESTART CREATE,ENDRBA=063000

```

Las cuestiones que debe tener en cuenta para reiniciar el iniciador de canal en el sitio de recuperación son similares a las que plantean al utilizar ARM para reiniciar el iniciador de canal en una imagen de z/OS diferente. Consulte [“Utilización de ARM en una red de IBM MQ”](#) en la página 533 para obtener más información. La estrategia de recuperación también debe cubrir la recuperación de las bibliotecas del producto IBM MQ y los entornos de programación de aplicaciones que utilizan IBM MQ (CICS, por ejemplo).

Otras funciones del programa de utilidad de inventario de registro de cambios (CSQJU003) también se pueden utilizar en escenarios de recuperación tras desastre. La función HIGHRBA permite la actualización de los valores de la RBA más alta grabada y la RBA más alta descargada dentro del conjunto de datos de rutina de carga. La función CHECKPT permite la adición de nuevos registros de cola de punto de comprobación o la supresión de registros de cola de punto de comprobación existentes en el BSDS.

**Atención: Estas funciones pueden afectar a la integridad de los datos de IBM MQ.** Utilícelas sólo en casos de recuperación tras desastre bajo la dirección del personal de servicio de IBM.

### Técnicas de copia rápida

Si se realizan copias de todos los conjuntos de páginas y los registros mientras el gestor de colas está bloqueado, las copias serán un conjunto coherente que puede utilizarse para reiniciar el gestor de colas en un sitio alternativo. Las copias suelen permitir un reinicio mucho más rápido del gestor de colas, ya que hay una recuperación desde soporte de almacenamiento mínima a realizar.

Utilice el mandato SUSPEND QMGR LOG para bloquear el gestor de colas. Este mandato desecha las agrupaciones de almacenamiento intermedio en los conjuntos de páginas, establece un punto de comprobación, y detiene cualquier actividad de grabación de registro posterior. Una vez que se ha suspendido la actividad de grabación de registro, el gestor de colas queda efectivamente bloqueado hasta que emita un mandato RESUME QMGR LOG. Mientras el gestor de colas está bloqueado, los conjuntos de páginas y los registros se pueden copiar.

Mediante el uso de herramientas de copia como FLASHCOPY o SNAPSHOT para copiar rápidamente los conjuntos de páginas y registros, el tiempo durante el cual el gestor de colas está bloqueado se puede reducir al mínimo.

No obstante, dentro de un grupo de compartición de colas, el mandato SUSPEND QMGR LOG podría no ser una buena solución. Para ser eficaces, las copias de los registros deben contener todas ellas el mismo punto en el tiempo para recuperación, lo que significa que el mandato SUSPEND QMGR LOG debe emitirse en todos los gestores de colas del grupo de compartición de colas simultáneamente y, por lo tanto, todo el grupo de compartición de colas quedará bloqueado durante algún tiempo.

## Recuperación de un grupo de compartición de colas

En el caso de un desastre en el sitio principal, puede reiniciar un grupo de compartición de colas en un sitio remoto utilizando conjuntos de datos de copia de seguridad del sitio principal. Para recuperar un grupo de compartición de colas, debe coordinar la recuperación en todos los gestores de colas del grupo de compartición de colas, y coordinarla con otros recursos, principalmente Db2. Este apartado describe detalladamente estas tareas.

- [Recuperación desde soporte de almacenamiento de estructuras CF](#)
- [Copia de seguridad del grupo de compartición de colas en el sitio principal](#)
- [Recuperación de un grupo de compartición de colas en el sitio alternativo](#)

### **Recuperación desde soporte de almacenamiento de estructuras CF**

La recuperación desde soporte de almacenamiento de una estructura CF que se utiliza para mantener mensajes persistentes en una cola compartida se basa en tener una copia de seguridad del soporte que la aplicación de actualizaciones registradas pueda recuperar hacia adelante. Realice copias de seguridad de las estructuras CF periódicamente mediante el mandato MQSC BACKUP CFSTRUCT. Todas las actualizaciones de colas compartidas (llamadas MQGET y MQPUT) se graban en el registro del gestor de colas donde se realiza la actualización. Para realizar la recuperación desde soporte de almacenamiento de una estructura CF debe aplicar actualizaciones registradas en esa copia de seguridad de los registros de todos los gestores de colas que han utilizado esa estructura CF. Cuando se utiliza el mandato MQSC RECOVER CFSTRUCT, automáticamente IBM MQ fusiona los registros de los gestores de colas relevantes y aplica las actualizaciones a la copia de seguridad más reciente.

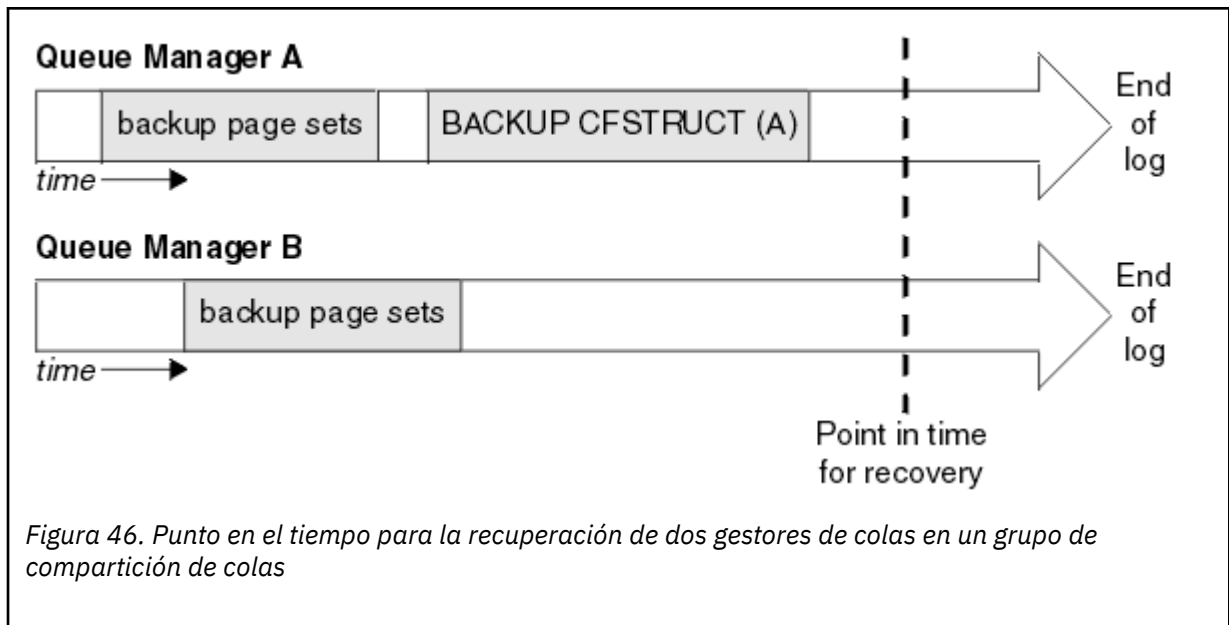
La copia de seguridad de la estructura CF se graba en el registro del gestor de colas que ha procesado el mandato BACKUP CFSTRUCT, por lo que no hay conjuntos de datos adicionales que se deban recopilar y transportar al sitio alternativo.

### **Copia de seguridad del grupo de compartición de colas en el sitio principal**

En el sitio principal, se debe establecer un conjunto coherente de copias de seguridad de forma regular, que se pueda utilizar en el caso de desastre para reconstruir el grupo de compartición de colas en un sitio alternativo. Para un único gestor de colas, la recuperación puede ser en un momento dado arbitrario, normalmente el final de los registros disponibles en el sitio remoto. Sin embargo, cuando se han almacenado mensajes persistentes en una cola compartida, los registros de todos los gestores de colas del grupo de compartición de colas se deben fusionar para recuperar colas compartidas, ya que cualquier gestor de colas del grupo de compartición de colas puede haber realizado actualizaciones (llamadas MQPUT o MQGET en la cola).

Para la recuperación de un grupo de compartición de colas, debe establecer un punto en el tiempo que esté dentro del intervalo de registro de los datos de registro de todos los gestores de colas. Sin embargo, como sólo puede recuperar **hacia adelante** soporte desde el registro, este punto en el tiempo debe ser posterior a la emisión del mandato BACKUP CFSTRUCT y posterior a la realización de cualquier copia de seguridad de conjunto de páginas. (Normalmente, el punto en el tiempo para la recuperación puede corresponder al final de un día o semana laboral.)

El diagrama siguiente muestra las líneas de tiempo para dos gestores de colas en un grupo de compartición de colas. Para cada gestor de colas, se realizan copias de seguridad difusas de conjuntos de páginas (consulte [Método 2: Copia de seguridad difusa](#)). En el gestor de colas A, se emite un mandato BACKUP CFSTRUCT. Más tarde, se emite un mandato ARCHIVE LOG en cada gestor de colas para truncar el registro activo y copiarlo en soporte fuera de línea del gestor de colas, que puede ser transportado al sitio alternativo. El final del registro identifica el momento en que se ha emitido el mandato ARCHIVE LOG y, por lo tanto, marca el alcance de los datos de registro que normalmente están disponibles en el sitio alternativo. El punto en el tiempo para la recuperación debe ir entre el final de cualquier copia de seguridad de conjunto de páginas o de estructura CF, y el primer final de registro disponible en el sitio alternativo.



IBM MQ registra información relacionada con las copias de seguridad de estructuras CF en una tabla de Db2. En función de sus necesidades, es posible que desee coordinar el punto en el tiempo para la recuperación de IBM MQ con el de Db2, o puede que sea suficiente hacer una copia de la tabla MQ CSQ.ADMIN\_B\_STRBACKUP de IBM MQ después de que hayan finalizado los mandatos BACKUP CFSTRUCT.

Para prepararse para una recuperación:

1. Cree copias de seguridad de conjuntos de páginas para cada gestor de colas del grupo de compartición de colas.
2. Emita un mandato BACKUP CFSTRUCT para cada estructura CF con el atributo RECOVER(YES). Puede emitir estos mandatos desde un único gestor de colas o desde diferentes gestores de colas dentro del grupo de compartición de colas para equilibrar la carga de trabajo.
3. Una vez completadas todas las copias de seguridad, emita un mandato ARCHIVE LOG para cambiar el registro activo y crear copias de los registros y los BSDS de cada gestor de colas en el grupo de compartición de colas.
4. Transporte las copias de seguridad de conjuntos de páginas, los registros archivados, el BSDS archivado de todos los gestores de colas del grupo de compartición de colas, y la información de copia de seguridad de Db2 elegida, fuera del sitio.

### Recuperación de un grupo de compartición de colas en el sitio alternativo

Para poder recuperar el grupo de compartición de colas, debe preparar el entorno:

1. Si dispone de información antigua en el recurso de acoplamiento de inicios de práctica cuando instaló el grupo de compartición de colas, primero tiene que borrarla:

**Nota:** Si no dispone de información antigua en el recurso de acoplamiento, puede omitir este paso.

- a. Emita el siguiente mandato de z/OS para ver las estructuras CF para este grupo de compartición de colas:

```
D XCF,STRUCTURE,STRNAME= qsgname
```

- b. Para todas las estructuras que empiezan por el nombre del grupo de compartición de colas, utilice el mandato z/OS SETXCF FORCE CONNECTION para forzar la conexión de esas estructuras:

```
SETXCF FORCE,CONNECTION,STRNAME= strname,CONNAME=ALL
```

- c. Suprima todas las estructuras CF utilizando el siguiente mandato para cada estructura:

```
SETXCF FORCE,STRUCTURE,STRNAME= strname
```

2. Restaure los sistemas Db2 y los grupos de compartición de datos.
3. Recupere la tabla CSQ.ADMIN\_B\_STRBACKUP para que contenga información sobre las copias de seguridad de estructura más recientes realizadas en el sitio principal.

**Nota:** Es importante que la tabla STRBACKUP contenga la información de copias de seguridad de estructura más reciente. La información de copias de seguridad de estructura más antigua podría requerir conjuntos de datos que se hayan descartado como resultado de la información proporcionada por un mandato DISPLAY USAGE TYPE(DATASET) reciente, lo que significaría que la estructura CF recuperada no contendría información exacta.

4. Ejecute el mandato ADD QMGR del programa de utilidad CSQ5PQSG para cada gestor de colas del grupo de compartición de colas. Esto restaurará la entrada de grupo XCF para cada gestor de colas.

Cuando ejecuta el programa de utilidad en este escenario, es normal que se muestren los siguientes mensajes:

```
CSQU566I Unable to get attributes for admin structure, CF not found
or not allocated
CSQU546E Unable to add QMGR queue_manager_name entry,
already exists in DB2 table CSQ.ADMIN_B_QMGR
CSQU148I CSQ5PQSG Utility completed, return code=4
```

Para recuperar los gestores de colas del grupo de compartición de colas:

1. Defina nuevos conjuntos de datos de conjunto de páginas y cárguelos con los datos de las copias de los conjuntos de páginas del sitio primario.
2. Defina nuevos conjuntos de datos de registro activo.
3. Defina un nuevo conjunto de datos BSDS y utilice el mandato REPRO de los Servicios de método de acceso para copiar el BSDS archivado más reciente en él.
4. Utilice el programa de utilidad imprimir correlación de registros, CSQJU004, para imprimir información de este BSDS más reciente. En el momento en que se archivó este BSDS, el registro archivado más reciente disponible se hubiera truncado como un registro activo y no aparece como un registro archivado. Anote los valores STARTRBA, STARTLRSN, ENDRBA y ENDLRSN de este registro.
5. Utilice el programa de utilidad de inventario de registro de cambios, CSQJU003, para registrar este conjunto de datos de registro de archivado más reciente en el BSDS que acaba de restaurar, utilizando los valores que ha anotado en el Paso “4” en la página 527.
6. Utilice la opción DELETE de CSQJU003 para eliminar toda la información de registro activo del BSDS.
7. Utilice la opción NEWLOG de CSQJU003 para añadir registros activos al BSDS, no especifique la STARTRBA o ENDRBA.
8. Calcule el *recoverylsn* para el grupo de compartición de colas. El *recoverylsn* es el más bajo de los ENDLRSNs en todos los gestores de colas del grupo de compartición de colas (como se ha registrado en el paso “4” en la página 527), menos 1. Por ejemplo, si hay dos gestores de colas en el grupo de compartición de colas, y el ENDLRSN para uno de ellos es B713 3C72 22C5, y para el otro es B713 3D45 2123, el *recoverylsn* es B713 3C72 22C4.
9. Utilice CSQJU003 para añadir un registro de control de reinicio para el BSDS. Especifique:



```
CRESTART CREATE,ENDLRSN= recoverylrsn
```

donde *recoverylrsn* es el valor que ha anotado en el paso “8” en la página 527.

El BSDS ahora describe todos los registros activos como vacíos, todos los registros de archivados que tiene disponibles, y ningún punto de comprobación después del final de los registros.

Debe añadir el registro CRESTART al BSDS para cada gestor de colas en el grupo de compartición de colas.

10. Reinicie cada gestor de colas del grupo de compartición de colas con el mandato START QMGR. Durante la inicialización, se emite un mensaje de respuesta del operador como el siguiente:

```
CSQJ245D +CSQ1 RESTART CONTROL INDICATES TRUNCATION AT RBA highrba.  
REPLY Y TO CONTINUE, N TO CANCEL
```

Responda Y para iniciar el gestor de colas. El gestor de colas se inicia, y recupera datos hasta la ENDRBA especificada en la sentencia CRESTART.

Para IBM WebSphere MQ 7.0.1 y posteriores, el primer gestor de colas que se inicia puede reconstruir las particiones de estructura de administración para los demás miembros del grupo de compartición de colas, así como la suya propia, y ya no es necesario reiniciar cada gestor de colas del grupo de compartición de colas en esta etapa.

11. Cuando se hayan reconstruido los datos de estructura de administración para todos los gestores de colas, emita un mandato RECOVER CFSTRUCT para cada estructura de aplicación CF.

Si emite el mandato RECOVER CFSTRUCT para todas las estructuras en un único gestor de colas, el proceso de fusión de registros sólo se realiza una vez, por lo que es más rápido que emitir el mandato en un gestor de colas diferente para cada estructura CF, donde cada gestor de colas tiene que realizar el paso de fusionar los registros.

Cuando se utiliza el proceso de reinicio condicional en un grupo de compartición de colas, los gestores de colas de IBM WebSphere MQ 7.0.1 y posteriores, que realizan la reconstrucción de administración de iguales, comprueban que los BSDS de iguales contienen el mismo LRSN CRESTART que el suyo propio. Esto es para asegurar la integridad de la estructura de administración reconstruida. Por eso es importante reiniciar otros iguales en el grupo de compartición de colas (QSG), para que puedan procesar su propia información CRESTART, antes del próximo reinicio incondicional de cualquier miembro del grupo.

## Utilización de la duplicación de disco

Muchas instalaciones ahora utilizan tecnologías de duplicación de disco, tales como IBM Metro Mirror (anteriormente PPRC), para realizar copias síncronas de los conjuntos de datos en un sitio alternativo. En estas situaciones, muchos de los pasos detallados resultan innecesarios, ya que los conjuntos de páginas y registros de IBM MQ en el sitio alternativo son efectivamente idénticos a los del sitio principal. Cuando se utilizan estas tecnologías, los pasos para reiniciar un grupo de compartición de colas en un sitio alternativo se pueden resumir como:

- Borrar las estructuras CF de IBM MQ en el sitio alternativo. (A menudo, contienen información residual de cualquier procedimiento anterior de recuperación tras desastre).
- Restaurar los sistemas Db2 y todas las tablas de la base de datos utilizadas por el grupo de compartición de colas de IBM MQ.
- Reiniciar los gestores de colas. En versiones anteriores a IBM WebSphere MQ 7.0.1, es necesario reiniciar cada gestor de colas definido en el grupo de compartición de colas, ya que cada cola recupera su propia partición de la estructura de administración durante el reinicio del gestor de colas. Después de que se ha reiniciado cada gestor de colas, aquellos que no están en sus LPAR de inicio se pueden concluir de nuevo. Para IBM WebSphere MQ 7.0.1 y posteriores, el primer gestor de colas que se inicia recompila las particiones de la estructura de administración para los demás miembros del grupo de



compartición de colas, así como la suya propia, y ya no es necesario reiniciar cada gestor de colas del grupo de compartición de colas.

- Una vez se ha reconstruido la estructura de administración, recuperar las estructuras de aplicación.

**V 9.1.2** IBM MQ 9.1.2, y posteriores, admite el uso de zHyperWrite al escribir en registros activos duplicados mediante Metro Mirror. zHyperWrite puede ayudar a reducir el impacto sobre el rendimiento del uso de Metro Mirror; consulte [Utilización de Metro Mirror con IBM MQ](#) si desea más información.

### **z/OS** *Reinicialización de un gestor de colas*

Si el gestor de colas ha terminado de forma anómala, es posible que no pueda reiniciarlo. Esto podría deberse a que los conjuntos de páginas o los registros se han perdido, truncado o dañado. Si sucede esto, probablemente, tendrá que reinicializar el gestor de colas (realizar un arranque en frío).

## **Atención**

**Sólo realice un arranque en frío si no puede reiniciar el gestor de colas de ninguna otra forma.**

Realizar un arranque en frío le permite recuperar el gestor de colas y las definiciones de objeto; **no** podrá recuperar los datos de mensaje. Compruebe que no funciona ninguno de los escenarios de reinicio descritos en este tema antes de hacerlo.

Cuando haya reiniciado, todos los objetos de IBM MQ están definidos y disponibles para su uso, pero no hay datos de mensaje.

**Nota:** No reinicialice un gestor de colas mientras forme parte de un clúster. En primer lugar, debe eliminar el gestor de colas del clúster (mediante los mandatos RESET CLUSTER en los otros gestores de colas del clúster) y, a continuación, reinicialícelo y, finalmente, vuélvalo a introducir en el clúster como un gestor de colas nuevo.

Esto se debe a que durante la reinicialización, el identificador del gestor de colas (QMID) se modifica, de forma que cualquier objeto de clúster con el identificador del gestor de colas antiguo se debe eliminar del clúster.

Si desea información adicional, consulte las siguientes secciones:

- [Reinicialización de un gestor de colas que no está en un grupo de compartición de colas](#)
- [Reinicialización de gestores de colas de un grupo de compartición de colas](#)

## **Reinicialización de un gestor de colas que no está en un grupo de compartición de colas**

Para reinicializar un gestor de colas, siga este procedimiento:

1. Prepare las sentencias de definición de objeto que se vayan a utilizar cuando reinicie el gestor de colas. Para hacer esto, puede:
  - Si está disponible el conjunto de páginas cero, utilice la función SDEFS de CSQUTIL (consulte [Creación de una lista de mandatos define de IBM MQ](#)). Debe obtener definiciones para todos los tipos de objeto (objetos de información de autenticación, estructuras CF, canales, listas de nombres, procesos, colas y clases de almacenamiento).
  - Si no está disponible el conjunto de páginas cero, utilice las definiciones de la última vez que hizo una copia de seguridad de las definiciones de objeto.
2. Redefina los conjuntos de datos del gestor de colas (no haga esto hasta que haya completado el paso “1” en la página 529).

Para obtener más información, consulte las secciones [Creación de la rutina de carga y los conjuntos de datos de registro](#) y [Definición de los conjuntos de páginas](#).

3. Reinicie el gestor de colas utilizando los conjuntos de datos de registro, BSDS y los conjuntos de páginas recién definidos e inicializados. Utilice las sentencias de entrada de definición de objeto que

ha creado en el paso “1” en la [página 529](#) como entrada en el conjunto de datos de entrada de inicialización de CSQINP2.

## Reinicialización de gestores de colas de un grupo de compartición de colas

En un grupo de compartición de colas, reiniciar un gestor de colas es más complejo. Tal vez sea necesario reiniciar uno o más gestores de colas debido a problemas de registro o del conjunto de páginas, pero también puede que haya problemas para solucionar con Db2 o con el recurso de acoplamiento. Debido a esto, existen una serie de alternativas:

### Arranque en frío

Reiniciar todo el grupo de compartición de colas completo implica forzar todas las estructuras de recursos de acoplamiento, borrar todas las definiciones de objeto para el grupo de compartición de colas de Db2, suprimir o redefinir los registros y BSDS y formatear los conjuntos de páginas para todos los gestores de colas del grupo de compartición de colas.

### Definiciones compartidas retenidas

Suprime o redefine los registros y BSDS, dé formato a los conjuntos de páginas de todos los gestores de colas del grupo de compartición de colas y fuerce todas las estructuras de recursos de acoplamiento. Durante el reinicio, se habrán borrado todos los mensajes. Los gestores de colas recrean los objetos COPY que corresponden a los objetos GROUP que siguen existiendo en la base de datos Db2. Las colas compartidas siguen existiendo y se pueden utilizar.

### Gestor de colas único reinicializado

Suprime o redefine los registros y BSDS y dé formato a los conjuntos de páginas para el gestor de colas único (así se suprimen todos sus objetos y mensajes privados). Durante el reinicio, el gestor de colas recrea los objetos COPY que corresponden a los objetos GROUP que siguen existiendo en la base de datos Db2. Las colas compartidas siguen existiendo, al igual que los mensajes incluidos en las mismas, y se pueden utilizar.

### Recuperación puntual de un grupo de compartición de colas

Se trata de un escenario de recuperación tras desastre de sitio alternativo.

Los objetos compartidos se recuperan en el punto en el tiempo alcanzado por la recuperación de Db2 (que se describe en [Un sistema Db2 falla](#)). Los gestores de colas se pueden recuperar en un punto en el tiempo que se consigue a partir de las copias de seguridad disponibles en el sitio alternativo.

Los mensajes persistentes se pueden utilizar en los grupos de compartición de colas y se pueden recuperar mediante el mandato MQSC RECOVER CFSTRUCT. Tenga en cuenta que este mandato recupera la hora de la anomalía. Sin embargo, no hay ninguna recuperación de los mensajes de cola compartidos no persistentes; se pierden, a menos que haya hecho copias de seguridad independientemente utilizando la función COPY del programa de utilidad CSQUTIL.

No es necesario intentar restaurar cada gestor de colas en el mismo punto en el tiempo porque no hay interdependencias entre los objetos locales en distintos gestores de colas (que son lo que se está recuperando realmente), y la resincronización del gestor de colas con Db2 durante el reinicio crea o suprime objetos COPY según sea necesario en un gestor de colas para cada gestor de colas.

## Utilización del Gestor de reinicio automático (ARM) de z/OS

Utilice este tema para comprender cómo puede utilizar el ARM para reiniciar automáticamente los gestores de colas.

Este apartado contiene información sobre los siguientes temas:

- “¿Qué es el ARM?” en la [página 531](#)
- “Políticas de ARM” en la [página 531](#)
- “Utilización de ARM en una red de IBM MQ” en la [página 533](#)

## ¿Qué es el ARM?

El Gestor de reinicio automático (ARM) de z/OS es una función de recuperación de z/OS que puede mejorar la disponibilidad de los gestores de colas. Cuando falla un trabajo o una tarea, o el sistema en el que se ejecuta falla, el ARM puede reiniciar el trabajo o la tarea sin la intervención del operador.

Si un gestor de colas o un iniciador de canal ha fallado, el ARM lo reinicia en la misma imagen de z/OS. Si z/OS, y por lo tanto un grupo completo de subsistemas relacionados de aplicaciones ha fallado, el ARM puede reiniciar automáticamente todos los sistemas que han fallado, en un orden predefinido, en otra imagen de z/OS dentro del sysplex. Esto se denomina *reinicio entre sistemas*.

Reinicie el iniciador de canal por ARM sólo en circunstancias excepcionales. Si el gestor de colas es reiniciado por el ARM, reinicie el iniciador de canal desde el conjunto de datos de inicialización de CSQINP2 (consulte [“Utilización de ARM en una red de IBM MQ”](#) en la página 533).

Puede utilizar el ARM para reiniciar un gestor de colas en una imagen de z/OS distinta dentro del sysplex en el caso de una anomalía de z/OS. Las implicaciones de red del reinicio de ARM de IBM MQ en una imagen de z/OS distinta se describen en [“Utilización de ARM en una red de IBM MQ”](#) en la página 533.

Para habilitar el reinicio automático:

- Configure un conjunto de datos de pareja de ARM.
- Defina las acciones de reinicio automático que desea que realice z/OS en una *política de ARM*.
- Inicie la política del ARM.

Asimismo, IBM MQ se debe registrar en el ARM durante el inicio (esto se produce automáticamente).

**Nota:** Si desea reiniciar los gestores de colas en distintas imágenes de z/OS automáticamente, debe definir cada gestor de colas como un subsistema en cada imagen de z/OS en la que se pueda reiniciar dicho gestor de colas, con un nombre de subsistema exclusivo de cuatro caracteres a nivel de sysplex.

### Conjuntos de datos de pareja de ARM

Asegúrese de que define los conjuntos de datos de acoplamiento necesarios para ARM y de que están en línea y activos antes de iniciar los gestores de colas para los cuales desea soporte de ARM. El registro de ARM automático de IBM MQ falla si los conjuntos de datos de acoplamiento no están disponibles durante el inicio del gestor de colas. En esta situación, IBM MQ presupone que la ausencia del conjunto de datos de acoplamiento significa que no desea el soporte de ARM y la inicialización continúa.

Consulte [z/OS MVS Setting up a Sysplex](#) para obtener información sobre los conjuntos de datos de acoplamiento de ARM.

### Políticas de ARM

Las políticas del Gestor de reinicio automático son reglas definidas por el usuario que controlan funciones de ARM que pueden controlar cualquier reinicio de un gestor de colas.

Las funciones de ARM se controlan mediante una *política ARM* definida por el usuario. Cada imagen de z/OS que ejecute una instancia de gestor de colas que vaya a ser reiniciada con ARM debe estar conectada a un conjunto de datos de acoplamiento de ARM con una política de ARM activa.

IBM proporciona una política de ARM predeterminada. Puede definir nuevas políticas o alterar temporalmente los valores predeterminados de política utilizando el programa de utilidad de datos administrativos de (IXCMIAPU) que se proporciona con z/OS. [z/OS MVS Setting up a Sysplex](#) describe este programa de utilidad e incluye detalles completos sobre cómo definir una política de ARM.

La [Figura 47](#) en la [página 532](#) muestra un ejemplo de una política de ARM. Esta política de ejemplo reinicia cualquier gestor de colas dentro de un sysplex, tanto si el gestor de colas ha fallado como si todo un sistema ha fallado.

```

//IXCMIAPU EXEC PGM=IXCMIAPU,REGION=2M
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DATA TYPE(ARM)
DEFINE POLICY NAME(ARMPOL1) REPLACE(YES)
RESTART_GROUP(DEFAULT)
ELEMENT(*)
RESTART_ATTEMPTS(0) /* Jobs not to be restarted by ARM */
RESTART_GROUP(GROUP1)
ELEMENT(SYSMQMGRMQ*) /* These jobs to be restarted by ARM */
/*

```

Figura 47. Política de ARM de ejemplo

Para obtener más información, consulte:

- [Definición de una política de ARM](#)
- [Activación de una política de ARM](#)
- [Registro con ARM](#)

## Definición de una política de ARM

Configure la política de ARM de la siguiente manera:

- Defina RESTART\_GROUPS para cada instancia de gestor de colas que también contenga cualquier subsistema CICS o IMS que se conecte a dicha instancia del gestor de colas. Si utiliza un convenio de denominación de subsistemas, es posible que pueda utilizar los caracteres comodín '?' y '\*' en los nombres de elemento para definir RESTART\_GROUPS con un esfuerzo mínimo de definición.
- Especifique TERMTYPE (ELEMTERM) para los iniciadores de canal para indicar que sólo se reiniciarán si el iniciador de canal ha fallado y la imagen de z/OS no ha fallado.
- Especifique TERMTYPE(ALLTERM) para los gestores de colas para indicar que se reiniciarán si el gestor de colas ha fallado o la imagen de z/OS ha fallado.
- Especifique RESTART\_METHOD(BOTH, PERSIST) tanto para los gestores de colas como para los iniciadores de canal. Esto indica a ARM que reinicie utilizando el JCL que ha guardado (después de la resolución de símbolos del sistema) durante el último inicio. Indica al ARM que haga esto independientemente de si el elemento individual ha fallado o la imagen de z/OS ha fallado.
- Acepte los valores predeterminados para todas las demás opciones de la política de ARM.

## Activación de una política de ARM

Para iniciar la política de gestión de reinicio automático, emita el siguiente mandato de z/OS:

```
SETXCF START,POLICY,TYPE=ARM,POLNAME= mypol
```

Cuando la política se ha iniciado, todos los sistemas conectados al conjunto de datos de acoplamiento de ARM utilizan la misma política activa.

Utilice el mandato STOP SETXCF para inhabilitar reinicios automáticos.

## Registro con ARM

IBM MQ se registra automáticamente como un *elemento ARM* durante el inicio del gestor de colas (sujeto a la disponibilidad de ARM). Se desregistra durante su fase de conclusión, a menos que se le solicite no hacerlo.

En el momento del inicio, el gestor de colas determina si ARM está disponible. Si lo está, IBM MQ se registra utilizando el nombre SYSMQMGR *ssid*, donde *ssid* es el nombre del gestor de colas de cuatro caracteres, y SYSMQMGR es el tipo de elemento.

Los mandatos STOP QMGR MODE(QUIESCE) y STOP QMGR MODE(FORCE) anulan el registro del gestor de colas del ARM (si se registró en el ARM durante el inicio). Esto impide que ARM reinicie este gestor de colas. El mandato STOP QMGR MODE(RESTART) no anula el registro del gestor de colas del ARM, por lo que es elegible para el reinicio automático inmediato.

Cada espacio de direcciones del iniciador de canal determina si ARM disponible, y si lo está, se registra con el nombre de elemento SYSMQCH*ssid*, donde *ssid* es el nombre del gestor de colas, y SYSMQCH es el tipo de elemento.

El iniciador de canal siempre se desregistra del ARM cuando se detiene con normalidad, y permanece registrado sólo si finaliza de forma anómala. El iniciador de canal siempre se desregistra si el gestor de colas falla.

## **Utilización de ARM en una red de IBM MQ**

Puede configurar el gestor de colas de forma que los iniciadores de canal y los escuchas asociados se inician automáticamente cuando se reinicia el gestor de colas.

Para garantizar el reinicio totalmente automático del gestor de colas en la misma imagen de z/OS para ambos protocolos de comunicación, LU 6.2 y TCP/IP:

- Inicie los escuchas automáticamente añadiendo el mandato START LISTENER apropiado al conjunto de datos CSQINPX.
- Inicie el iniciador de canal automáticamente añadiendo el mandato START CHINIT apropiado al conjunto de datos CSQINP2.

Para reiniciar un gestor de colas con TCP/IP o LU6.2, consulte

- [“Reinicio en una imagen de z/OS diferente con TCP/IP” en la página 533](#)
- [“Reinicio en una imagen de z/OS diferente con LU 6.2” en la página 535](#)

Consulte [Tarea 13: Personalizar los conjuntos de datos de entrada de inicialización](#) para obtener información sobre los conjuntos de datos CSQINP2 y CSQINPX.

## **Reinicio en una imagen de z/OS diferente con TCP/IP**

Si está utilizando TCP/IP como el protocolo de comunicaciones y está utilizando las direcciones IP virtuales, puede configurar estos para recuperar en otras imágenes de z/OS, lo que permite a los canales conectarse a dicho gestor de colas para reconectarse sin ningún cambio. De lo contrario, puede reasignar una dirección TCP/IP después de mover un gestor de colas a una imagen de z/OS diferente, solo si utiliza clústeres o si se está conectando a un grupo de compartición de colas utilizando el nombre de grupo lógico DNS (sistema de nombres de dominio) dinámico WLM.

- [Al utilizar la agrupación en clúster](#)
- [Al conectarse a un grupo de compartición de colas](#)

### **Al utilizar la agrupación en clúster**

El Gestor de reinicio automático (ARM) de z/OS responde a una anomalía del sistema reiniciando el gestor de colas en una imagen de z/OS diferente en el mismo sysplex; este sistema tiene una dirección TCP/IP diferente para la imagen de z/OS original. A continuación se explica cómo puede utilizar los clústeres de IBM MQ para reasignar una dirección TCP/IP del gestor de colas, después de que el reinicio de ARM lo haya movido a una imagen de z/OS distinta.

Cuando un gestor de colas de cliente detecta la anomalía del gestor de colas (como una anomalía de canal), responde reasignando mensajes idóneas en su cola de transmisión de clúster a un gestor

de colas de servidor diferente que aloja una instancia distinta de la cola de clúster de destino. Sin embargo, no puede reasignar los mensajes enlazados con el servidor original mediante restricciones de afinidad, o los mensajes que son dudosos porque el gestor de colas de servidor ha no se realiza correctamente durante el proceso final de lotes. Para procesar estos mensajes, haga lo siguiente:

1. Asigne un nombre de canal de clúster receptor diferente y un puerto TCP/IP distinto a cada gestor de colas z/OS. Cada gestor de colas necesita un puerto diferente de forma que dos sistemas pueden compartir una sola pila TCP/IP en una imagen de z/OS. Uno de estos es el gestor de colas que originalmente se ejecuta en dicha imagen de z/OS y el otro es el gestor de colas que reiniciará el ARM en dicha imagen de z/OS después de una anomalía del sistema. Configure cada puerto en cada imagen de z/OS, para que el ARM pueda reiniciar cualquier gestor de colas en cualquier imagen de z/OS.
2. Cree una combinación del archivo de entrada de mandato de iniciador de canal diferente (CSQINPX) para cada gestor de colas y una imagen de z/OS, para que se pueda hacer referencia a la misma durante el arranque del iniciador de canal.

Cada archivo CSQINPX debe incluir un mandato START LISTENER PORT(puerto) específico para dicho gestor de colas y un mandato ALTER CHANNEL para un canal de clúster receptor específico para dicha combinación de gestor de colas e imagen de z/OS. El mandato ALTER CHANNEL debe definir el nombre de conexión en el nombre TCP/IP de la imagen de z/OS en la que se reinicia. Debe incluir el número de puerto específico del gestor de colas reiniciado como parte del nombre de conexión.

El JCL de inicio de cada gestor de colas puede tener un nombre de conjunto de datos fijo para este archivo CSQINPX y cada imagen de z/OS debe tener una versión diferente de cada archivo CSQINPX en un volumen DASD no compartido.

Si se produce un reinicio de ARM, IBM MQ anuncia la definición de canal modificada en el repositorio del clúster, que, a su vez, lo publica en todos los gestores de colas de cliente que han expresado su interés en el gestor de colas del servidor.

El gestor de colas de cliente trata la anomalía del gestor de colas de servidor como una anomalía de canal e intentar reiniciar el canal fallido. Cuando el gestor de colas de cliente conoce el nuevo nombre de conexión con el servidor, el reinicio del canal reconecta el gestor de colas de cliente con el gestor de colas de servidor reiniciado. El gestor de colas de cliente puede resincronizar sus mensajes, resolver cualquier mensaje pendiente en la cola de transmisión del gestor de colas de cliente y el proceso normal puede continuar.

### **Al conectarse a un grupo de compartición de colas**

Al conectarse a un grupo de compartición de colas a través de un nombre de grupo lógico del sistema de nombres (DNS) dinámico de TCP/IP, el nombre de conexión de la definición de canal especifica el nombre de grupo lógico del grupo de compartición de colas, no el nombre de host o la dirección IP de una máquina física. Cuando se inicia este canal, se conecta al DNS dinámico y se conecta a uno de los gestores de colas del grupo de compartición de colas. Este proceso se explica en [Configuración de la comunicación para IBM MQ for z/OS utilizando grupos de compartición de colas](#).

En el caso improbable de una anomalía de imagen, se produce una de las siguientes situaciones:

- Los gestores de colas en la imagen que no se realizan correctamente eliminan sus registros del DNS dinámico que se ejecute en el sysplex. El canal responde a la anomalía de conexión especificando el estado RETRYING y, a continuación, se conecta al DNS dinámico que se ejecuta en el sysplex. El DNS dinámico asigna la solicitud de entrada a uno de los miembros restantes del grupo de compartición de colas que sigue ejecutándose en las imágenes restantes.
- Si no hay ningún otro gestor de colas del grupo de compartición de colas activo y ARM reinicia el gestor de colas y el iniciador de canal en una imagen distinta, el escucha del grupo se registra con el DNS dinámico desde esta nueva imagen. Esto significa que el nombre de grupo lógico (del campo de nombre de conexión del canal) se conecta al DNS dinámico y, a continuación, se conecta al mismo gestor de colas, que ahora se ejecuta en una imagen diferente. No fue necesario ningún cambio en la definición de canal.

Para que se produzca este tipo de recuperación, se deben tener en cuenta los siguientes puntos:

- En z/OS, el DNS dinámico se ejecuta en una de las imágenes de z/OS del sysplex. Si esta imagen fallara, el DNS dinámico se debe configurar de forma que exista un servidor de nombres secundario activo en el sysplex, que actúe como alternativa al servidor de nombres primario. Encontrará información sobre los servidores DNS dinámicos primarios y secundarios en el manual [OS/390 SecureWay CS IP Configuration](#).
- El escucha del grupo TCP/IP se podría haber iniciado en una dirección IP particular que es posible que no esté disponible en esta imagen de z/OS. Si es así, es probable que se deba reiniciar el escucha en una dirección IP distinta en la nueva imagen. Si utiliza las direcciones IP virtuales, puede configurarlas para realizar la recuperación en otras imágenes de z/OS de manera que no es necesario ningún cambio en el mandato START LISTENER.

## Reinicio en una imagen de z/OS diferente con LU 6.2

Si sólo utiliza los protocolos de comunicación LU 6.2, lleve a cabo el siguiente procedimiento para habilitar la reconexión de red después de un reinicio automático de un gestor de colas en una imagen de z/OS diferente dentro del sysplex:

- Defina cada gestor de colas dentro del sysplex con un nombre de subsistema exclusivo.
- Defina cada iniciador de canal dentro del sysplex con un LUNAME único. Esto se especifica en los atributos del gestor de colas y, también, en el mandato START LISTENER.

**Nota:** LUNAME denomina una entrada en la tabla auxiliar de APPC, que, a su vez, correlaciona esto con el LUNAME real.

- Configure una tabla auxiliar de APPC compartida, a la que hace referencia cada imagen de z/OS dentro del sysplex. Esto debe contener una entrada para cada LUNAME del iniciador de canal. Consulte [z/OS MVS Planning: APPC/MVS Management](#) para obtener información al respecto.
- Configure un miembro APPCPMxx de SYS1.PARMLIB cada iniciador de canal dentro del sysplex para contener un LUADD para activar la entrada de la tabla auxiliar de APPC para dicho iniciador de canal. Estos miembros deben ser compartidos por cada imagen de z/OS. El miembro SYS1.PARMLIB apropiado se activa mediante un mandato z/OS SET APPC= xx, que se emite de forma automática durante el reinicio de ARM del gestor de colas (y su iniciador de canal en una imagen de z/OS distinta, tal como se describe en el siguiente texto).
- Utilice el atributo del gestor de colas LU62ARM para especificar el sufijo xx de este miembro SYS1.PARMLIB para cada iniciador de canal. Esto hace que el iniciador de canal emita el mandato z/OS SET APPC= xx para activar su LUNAME.

Defina la política de ARM para que reinicie el iniciador de canal sólo si no se realiza correctamente, mientras su imagen de z/OS permanece; el ID de usuario asociado al espacio de direcciones XCFAS debe estar autorizado para emitir el mandato de IBM MQ START CHINIT. No reinicie el iniciador de canal automáticamente si su imagen de z/OS también falla, en lugar de esto, utilice los mandatos de los conjuntos de datos CSQINP2 y CSQINPX para iniciar los escuchas y el iniciador de canal.

## Recuperar unidades de trabajo manualmente

Puede recuperar manualmente unidades de trabajo CICS, IMS, RRS u otros gestores de colas en un grupo de compartición de colas. Puede utilizar mandatos del gestor de colas para visualizar el estado de las unidades de trabajo asociadas a cada conexión con el gestor de colas.

Este tema contiene información sobre los temas siguientes:

- [“Visualizar conexiones y hebras” en la página 536](#)
- [“Recuperar de unidades de recuperación CICS manualmente” en la página 536](#)
- [“Recuperar de unidades de recuperación IMS manualmente” en la página 540](#)
- [“Recuperar unidades de recuperación RRS manualmente” en la página 542](#)



- [“Recuperación de unidades de recuperación en otro gestor de colas del grupo de compartición de colas” en la página 542](#)

## Visualizar conexiones y hebras

Puede utilizar el mandato `DISPLAY CONN` para obtener información sobre las conexiones a los gestores de colas y sus unidades de trabajo asociadas. Puede visualizar las unidades de trabajo activas para ver lo que está ocurriendo actualmente, o para ver lo que hay que terminar para que el gestor de colas pueda concluir, y puede visualizar las unidades de trabajo sin resolver para ayudar con la recuperación.

### Unidades de trabajo activas

Para visualizar sólo las unidades de trabajo activas, utilice

```
DISPLAY CONN(*) WHERE(UOWSTATE EQ ACTIVE)
```

### Unidades de trabajo sin resolver

Una unidad de trabajo sin resolver, también conocida como "hebra pendiente", es una que está en el segundo pase de la operación de confirmación en dos fases. Los recursos se mantienen en IBM MQ en su nombre. Para visualizar las unidades de trabajo sin resolver, utilice

```
DISPLAY CONN(*) WHERE(UOWSTATE EQ UNRESOLVED)
```

Es necesaria una intervención externa para resolver el estado de las unidades de trabajo sin resolver. Esto puede implicar únicamente iniciar el coordinador de recuperación (CICS, IMS o RRS) o puede implicar algo más, tal como se describe en las siguientes secciones.

## **Recuperar de unidades de recuperación CICS manualmente**

Este tema describe lo que sucede cuando el adaptador CICS se reinicia y también explica cómo manejar las unidades de recuperación sin resolver que puedan surgir.

## Qué sucede cuando el adaptador CICS se reinicia

Cada vez que una conexión se interrumpe, el adaptador tiene que pasar por una *fase de reinicio* durante el *proceso de reconexión*. La fase de reinicio resincroniza los recursos. La resincronización entre CICS e IBM MQ permite identificar y resolver las unidades de trabajo pendientes.

La causa de la resincronización puede ser:

- Una solicitud explícita del componente de gestión de colas distribuidas
- Una solicitud implícita cuando se establece una conexión con IBM MQ

Si la causa de la resincronización es la conexión a IBM MQ, la secuencia de sucesos es la siguiente:

1. El proceso de conexión recupera una lista de identificadores (ID) de unidades de trabajo (UOW) pendientes de IBM MQ.
2. Los ID de unidad de trabajo se muestran en la consola en los mensajes CSQC313I.
3. Los ID de unidad de trabajo se pasan a CICS.
4. CICS inicia una tarea de resincronización (CRSY) para cada ID de unidad de trabajo pendiente.
5. El resultado de la tarea para cada unidad de trabajo pendiente se muestra en la consola.

Debe comprobar los mensajes que se visualizan durante el proceso de conexión:

### **CSQC313I**

Indica que una unidad de trabajo está pendiente.



### CSQC400I

Identifica la unidad de trabajo y va seguido de uno de estos mensajes:

- CSQC402I o CSQC403I indica que la unidad de trabajo se ha resuelto satisfactoriamente (confirmada o restituida).
- CSQC404E, CSQC405E, CSQC406E o CSQC407E indica que la unidad de trabajo no se ha resuelto.

### CSQC409I

Indica que todas las unidades de trabajo se han resuelto satisfactoriamente.

### CSQC408I

Indica que no todas las unidades de trabajo se han resuelto satisfactoriamente.

### CSQC314I

Advierte que los ID de unidad de trabajo marcados con un asterisco \* no se resuelven automáticamente. Estas unidades de trabajo las debe resolver explícitamente el componente de colas distribuidas cuando se reinicie.

La [Figura 48](#) en la [página 537](#) muestra un ejemplo de una serie de mensajes de reinicio que se visualizan en la consola de z/OS.

```
CSQ9022I +CSQ1 CSQYASCP ' START QMGR' NORMAL COMPLETION
+CSQC323I VICIC1 CSQCQCON CONNECT received from TERMID=PB62 TRANID=CKCN
+CSQC303I VICIC1 CSQCCON CSQCSERV loaded. Entry point is 850E8918
+CSQC313I VICIC1 CSQCCON UOWID=VICIC1.A6E5A6F0E2178D25 is in doubt
+CSQC313I VICIC1 CSQCCON UOWID=VICIC1.A6E5A6F055B2AC25 is in doubt
+CSQC313I VICIC1 CSQCCON UOWID=VICIC1.A6E5A6EFFD60D425 is in doubt
+CSQC313I VICIC1 CSQCCON UOWID=VICIC1.A6E5A6F07AB56D22 is in doubt
+CSQC307I VICIC1 CSQCCON Successful connection to subsystem VC2
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008BAD18) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008BAA10) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008BA708) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CAE88) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CAB80) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CA878) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CA570) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CA268) connect
successful
+CSQC403I VICIC1 CSQCTRUE Resolved BACKOUT for
+CSQC400I VICIC1 CSQCTRUE UOWID=VICIC1.A6E5A6F0E2178D25
+CSQC403I VICIC1 CSQCTRUE Resolved BACKOUT for
+CSQC400I VICIC1 CSQCTRUE UOWID=VICIC1.A6E5A6F055B2AC25
+CSQC403I VICIC1 CSQCTRUE Resolved BACKOUT for
+CSQC400I VICIC1 CSQCTRUE UOWID=VICIC1.A6E5A6F07AB56D22
+CSQC403I VICIC1 CSQCTRUE Resolved BACKOUT for
+CSQC400I VICIC1 CSQCTRUE UOWID=VICIC1.A6E5A6EFFD60D425
+CSQC409I VICIC1 CSQCTRUE Resynchronization completed successfully
```

*Figura 48. Mensajes de reinicio de ejemplo*

El número total de mensajes CSQC313I debe ser igual al número total de mensajes CSQC402I y CSQC403I. Si los totales no son iguales, hay unidades de trabajo que el proceso de conexión no puede resolver. Esas unidades de trabajo que no se pueden resolver están causadas por problemas con CICS (por ejemplo un arranque en frío) o con IBM MQ, o por la gestión de colas distribuidas. Una vez que se hayan resuelto estos problemas, puede iniciar otra resincronización, desconectándose y luego volviendo a conectarse.

Como alternativa, puede resolver usted mismo cada unidad de trabajo pendiente mediante el mandato RESOLVE INDOUBT y el ID de unidad de trabajo que aparece en el mensaje CSQC400I. A continuación, debe iniciar una desconexión y una conexión para limpiar los *descriptores de unidad de recuperación* en

CICS. Debe saber el resultado correcto de la unidad de trabajo para resolver las unidades de trabajo manualmente.

Todos los mensajes que están asociados a unidades de trabajo sin resolver están bloqueados por IBM MQ y ninguna tarea de proceso por lotes, TSO o CICS puede acceder a ellos.

Si CICS falla y es necesario un reinicio de emergencia, *no* modifique el GENERIC APPLID del sistema CICS. Si lo hace y luego se vuelve a conectar a IBM MQ, la integridad de datos con IBM MQ no se puede garantizar. Esto es debido a que IBM MQ trata la nueva instancia de CICS como un CICS diferente (porque el APPLID es diferente). La resolución de unidades de trabajo pendientes se basa entonces en el registro de CICS incorrecto.

## Cómo resolver unidades de recuperación CICS manualmente

Si el adaptador termina de forma anómala, CICS e IBM MQ crean listas de unidades de trabajo pendientes de forma dinámica o durante el reinicio, dependiendo del subsistema que ha provocado la terminación anómala.

**Nota:** Si utiliza el programa de ejemplo DFH\$INDB para mostrar unidades de trabajo, puede encontrar que éste no siempre muestra las unidades de trabajo de IBM MQ correctamente.

Cuando CICS se conecta a IBM MQ, puede haber una o más unidades de recuperación que no se hayan resuelto.

Uno de los siguientes mensajes se envía a la consola:

- CSQC404E
- CSQC405E
- CSQC406E
- CSQC407E
- CSQC408I

Para obtener información detallada de lo que significan estos mensajes, consulte los [Mensajes de Puente y adaptador CICS](#).

CICS conserva detalles de las unidades de recuperación que no se resolvieron durante el inicio de la conexión. Una entrada se depura cuando ya no aparece en la lista presentada por IBM MQ.

Todas las unidades de recuperación que CICS no pueda resolver se deben resolver manualmente utilizando mandatos de IBM MQ. Este procedimiento manual rara vez se utiliza en una instalación, ya que sólo es necesario si errores operativos o problemas de software han impedido la resolución automática. *Todas las incoherencias encontradas durante la resolución de unidades de trabajo pendientes deben ser investigadas.*

Para resolver las unidades de recuperación:

1. Obtenga una lista de las unidades de recuperación de IBM MQ utilizando el siguiente mandato:

```
+CSQ1 DISPLAY CONN( * ) WHERE(UOWSTATE EQ UNRESOLVED)
```

Recibirá el siguiente mensaje:

```

CSQM201I +CSQ1 CSQMDRTC DISPLAY CONN DETAILS
CONN (BC85772CBE3E0001)
EXTCONN (C3E2D8C3C7D9F0F940404040404040)
TYPE (CONN)
CONNOPTS (
MQCNO_STANDARD_BINDING
)
UOWLOGDA (2005-02-04)
UOWLOGTI (10.17.44)
UOWSTDA (2005-02-04)
UOWSTTI (10.17.44)
UOWSTATE (UNRESOLVED)
NID (IYRCSQ1 .BC8571519B60222D)
EXTURID (BC8571519B60222D)
QMURID (0000002BDA50)
URTYPE (CICS)
USERID (MQTEST)
APPLTAG (IYRCSQ1)
ASID (0000)
APPLTYPE (CICS)
TRANSID (GP02)
TASKNO (0000096)
END CONN DETAILS

```

Para las conexiones CICS, el NID (identificador de red) consta del applid (identificador de aplicación) de CICS y de un número exclusivo que CICS proporciona en el momento en que se graban las entradas del registro de punto de sincronización. Este número exclusivo se almacena en registros que se graban tanto en el registro del sistema de CICS como en el registro de IBM MQ durante el tiempo de proceso de punto de sincronización. Este valor es conocido en CICS como la *señal de recuperación*.

2. Examine el registro de CICS para ver si hay entradas relacionadas con una determinada unidad de recuperación.

Busque un registro PREPARE para la instalación relacionada con la tarea donde el campo de señal de recuperación (JCSRMTKN) sea igual al valor obtenido del ID de red. IBM MQ proporciona el ID de red en la salida del mandato DISPLAY CONN.

El registro PREPARE en el registro de CICS para las unidades de recuperación proporciona el número de tarea de CICS. Todas las demás entradas del registro para esta tarea CICS se pueden localizar utilizando este número.

Puede utilizar el programa de utilidad de impresión de diario de CICS DFHJUP al examinar el registro. Para obtener información detallada sobre la utilización de este programa, consulte la publicación *CICS Operations and Utilities Guide*.

3. Examine el registro de IBM MQ para ver si hay registros con el NID relacionado con una determinada unidad de recuperación. A continuación, utilice el URID de este registro para obtener el resto de los registros de anotaciones para esta unidad de recuperación.

Al examinar el registro de IBM MQ, tenga en cuenta que el mensaje de inicio CSQJ001I de IBM MQ proporciona la RBA inicial para esta sesión.

Se puede utilizar el programa de utilidad imprimir registros de anotaciones (CSQ1LOGP) con este fin.

4. Si es necesario, realice la resolución de unidades de trabajo pendientes en IBM MQ.

Se puede ordenar a IBM MQ que realice la acción de recuperación para una unidad de recuperación utilizando un mandato de IBM MQ [RESOLVE INDOUBT](#).

Para recuperar todas las hebras asociadas a un *nombre-conexión* específico, utilice la opción NID(\*).

El mandato produce uno de los siguientes mensajes que muestran si la hebra se ha confirmado o restituído:

```
CSQV414I +CSQ1 THREAD network-id COMMIT SCHEDULED
CSQV415I +CSQ1 THREAD network-id ABORT SCHEDULED
```

Al realizar la resolución de unidades pendientes, CICS y el adaptador no están informados de los mandatos emitidos a IBM MQ para confirmar o restituir unidades de recuperación, ya que sólo los recursos de IBM MQ se ven afectados. No obstante, CICS mantiene detalles sobre las hebras pendientes que IBM MQ no ha podido resolver. Esta información se depura cuando la lista presentada está vacía o cuando la lista no incluye una unidad de recuperación de la que CICS tiene detalles.

## **Recuperar de unidades de recuperación IMS manualmente**

Este tema describe lo que sucede cuando el adaptador IMS se reinicia y también explica cómo manejar las unidades de recuperación sin resolver que puedan surgir.

### **Qué sucede cuando el adaptador IMS se reinicia**

Cada vez que se reinicia la conexión a IBM MQ, ya sea después de un reinicio del gestor de colas o después de un mandato de IMS /START SUBSYS, IMS inicia el siguiente proceso de resincronización:

1. IMS presenta la lista de identificadores (ID) de unidad de trabajo (UOW) que cree que están pendientes al adaptador IBM MQ IMS, uno a uno, con un parámetro de resolución de Confirmación o Restitución.
2. El adaptador IMS pasa la solicitud de resolución a IBM MQ y devuelve el resultado a IMS.
3. Después de haber procesado todas las solicitudes de resolución de IMS, el adaptador IMS obtiene de IBM MQ una lista de todas las unidades de trabajo que IBM MQ todavía mantiene pendientes y que fueron iniciadas por el sistema IMS. Estas se notifican al terminal maestro de IMS en el mensaje CSQQ008I.

**Nota:** Mientras una unidad de trabajo está pendiente, cualquier mensaje de IBM MQ asociado está bloqueado por IBM MQ y no está disponible para ninguna aplicación.

### **Cómo resolver unidades de recuperación IMS manualmente**

Cuando IMS se conecta a IBM MQ, IBM MQ puede tener una o más unidades de recuperación pendientes que no se han resuelto.

Si IBM MQ tiene unidades de recuperación pendientes que IMS no ha resuelto, se emite el siguiente mensaje en el terminal maestro de IMS:

```
CSQQ008I nn units of recovery are still in doubt in queue manager qmgr-name
```

Si se emite este mensaje, IMS se ha arrancado en frío o se ha iniciado con una cinta de registro incompleta. Este mensaje también se puede emitir si IBM MQ o IMS termina de forma anómala debido a un error de software u otra anomalía del subsistema.

Después de recibir el mensaje CSQQ008I:

- La conexión permanece activa.
- Las aplicaciones IMS todavía pueden acceder a los recursos de IBM MQ.
- Algunos recursos de IBM MQ permanecen bloqueados.

Si la hebra pendiente no se resuelve, las colas de mensajes de IMS se pueden empezar a llenar. Si las colas de IMS se llenan completamente, IMS termina. Debe ser consciente de esta posible dificultad, y debe supervisar IMS hasta que las unidades de recuperación pendientes se hayan resuelto completamente.

## Procedimiento de recuperación

Utilice el siguiente procedimiento para recuperar las unidades de trabajo IMS:

1. Fuerce el cierre del registro de IMS utilizando /SWI OLDS, y luego archive el registro de IMS. Utilice el programa de utilidad DFSERA10 para imprimir los registros de la cinta de registro de IMS anterior. Los registros de anotaciones de tipo X'3730' indican una solicitud de compromiso en dos fases y los registros de anotaciones de tipo X'38' indican una solicitud de terminación anormal. Anote la acción solicitada para la última transacción en cada región dependiente.
2. Ejecute el trabajo por lotes DL/I para restituir cada PSB afectado que no ha llegado a un punto de confirmación. El proceso puede tardar algún tiempo ya que todavía se están procesando transacciones. También puede bloquear una serie de registros, lo que podría afectar al resto del proceso y al resto de colas de mensajes.
3. Genere una lista de las unidades de recuperación pendientes de IBM MQ utilizando el siguiente mandato:

```
+CSQ1 DISPLAY CONN(*) WHERE(UOWSTATE EQ UNRESOLVED)
```

Recibirá el siguiente mensaje:

```
CSQM201I +CSQ1 CSQMDRTC DISPLAY CONN DETAILS
CONN(BC45A794C4290001)
EXTCONN(C3E2D8C3E2C5C3F240404040404040)
TYPE(CONN)
CONNOPTS(
MQCNO_STANDARD_BINDING
)
UOWLOGDA(2005-02-15)
UOWLOGTI(16.39.43)
UOWSTDA(2005-02-15)
UOWSTTI(16.39.43)
UOWSTATE(UNRESOLVED)
NID(IM8F .BC45A794D3810344)
EXTURID(
0000052900000000
)
QMURID(00000354B76E)
URTYPE(IMS)
USERID(STCPI)
APPLTAG(IM8F)
ASID(0000)
APPLTYPE(IMS)
PSTID(0004)
PSBNAME(GP01MPP)
```

Para IMS, el NID (identificador de red) consta del nombre de conexión IMS y de un número exclusivo proporcionado por IMS. Este valor es conocido en IMS como la *señal de recuperación*. Para obtener más información, consulte la publicación *IMS Customization Guide*.

4. Compare los NID (el IMSID más el OASN (número de planificación de aplicación de origen) en hexadecimal) que aparecen en los mensajes DISPLAY THREAD con los OASN (decimal de 4 bytes) mostrados en la salida de DFSERA10. Decida si desea confirmar o restituir.
5. Realice la resolución de unidades de trabajo pendientes en IBM MQ con el mandato RESOLVE INDOUBT, de la siguiente manera:

```
RESOLVE INDOUBT( connection-name )
ACTION(COMMIT|BACKOUT)
NID( network-id )
```

Para recuperar todas las hebras asociadas al *nombre-conexión*, utilice la opción NID(\*). El mandato da como resultado uno de los siguientes mensajes para indicar si la hebra se ha confirmado o restituido:

```
CSQV414I  THREAD network-id COMMIT SCHEDULED
CSQV415I  THREAD network-id BACKOUT SCHEDULED
```

Al realizar la resolución de unidades pendientes, IMS y el adaptador no están informados de los mandatos emitidos a IBM MQ para confirmar o restituir unidades de recuperación pendientes, ya que sólo los recursos de IBM MQ se ven afectados.

### **z/OS** **Recuperar unidades de recuperación RRS manualmente**

Lea este tema para comprender cómo determinar si hay unidades de recuperación RRS pendientes y cómo resolver manualmente dichas unidades de recuperación.

Cuando RRS se conecta a IBM MQ, IBM MQ puede tener una o más unidades de recuperación pendientes que no se han resuelto. Si IBM MQ tiene unidades de recuperación pendientes que RRS no ha resuelto, se emite uno de los siguientes mensajes en la consola de z/OS:

- CSQ3011I
- CSQ3013I
- CSQ3014I
- CSQ3016I

Tanto IBM MQ como RRS proporcionan herramientas para visualizar información sobre unidades de recuperación pendientes, y técnicas para resolverlas manualmente.

En IBM MQ utilice el mandato DISPLAY CONN para visualizar información sobre hebras de IBM MQ pendientes. La salida del mandato incluye identificadores (ID) de unidad de recuperación RRS para las hebras de IBM MQ que tienen RRS como coordinador. Esto se puede utilizar para determinar el resultado de la unidad de recuperación.

Utilice el mandato RESOLVE INDOUBT para resolver manualmente la hebra de IBM MQ pendiente. Este mandato se puede utilizar para confirmar o restituir la unidad de recuperación después de que haya determinado cuál es la decisión correcta.

### **z/OS** **Recuperación de unidades de recuperación en otro gestor de colas del grupo de compartición de colas**

Utilice este tema para identificar y recuperar manualmente las unidades de recuperación en otros gestores de colas de un grupo de compartición de colas.

Si un gestor de colas que es miembro de un grupo de compartición de colas falla y no se puede reiniciar, otros gestores de colas del grupo pueden realizar la recuperación de iguales y reemplazarlo. Sin embargo, el gestor de colas puede tener unidades de recuperación pendientes que no se pueden resolver mediante la recuperación de iguales porque la disposición final de esa unidad de recuperación sólo es conocida por el gestor de colas que ha fallado. Estas unidades de recuperación se resuelven cuando el gestor de colas finalmente se reinicia, pero hasta entonces, permanecen pendientes.

Esto significa que ciertos recursos (por ejemplo, los mensajes) pueden estar bloqueados, por lo que no están disponibles para otros gestores de colas del grupo. En esta situación, puede utilizar el mandato DISPLAY THREAD para visualizar estas unidades de trabajo en el gestor de colas inactivo. Si desea resolver estas unidades de recuperación manualmente para que los mensajes estén disponibles para otros gestores de colas del grupo, puede utilizar el mandato RESOLVE INDOUBT.

Cuando se emite el mandato DISPLAY THREAD para visualizar unidades de recuperación que están pendientes, se puede utilizar la palabra clave QMNAME para especificar el nombre del gestor de colas inactivo. Por ejemplo, si emite el mandato siguiente:

```
+CSQ1 DISPLAY THREAD(*) TYPE(INDOUBT) QMNAME(QM01)
```

Recibirá los siguientes mensajes:

```
CSQV436I +CSQ1 INDOUBT THREADS FOR QM01 -  
NAME   THREAD-XREF   URID NID  
USER1  0000000000000000000000000000 CSQ:0001.0  
USER2  0000000000000000000000000000 CSQ:0002.0  
DISPLAY THREAD REPORT COMPLETE
```

Si el gestor de colas especificado está activo, IBM MQ no devuelve información sobre las hebras pendientes, pero emite el siguiente mensaje:

```
CSQV435I CANNOT USE QMNAME KEYWORD, QM01 IS ACTIVE
```

Utilice el mandato de IBM MQ RESOLVE INDOUBT para resolver las hebras pendientes manualmente. Utilice la palabra clave QMNAME para especificar el nombre del gestor de colas inactivo en el mandato.

Este mandato se puede utilizar para confirmar o restituir la unidad de recuperación. El mandato resuelve solamente la parte compartida de la unidad de recuperación; los mensajes locales no se ven afectados y permanecen bloqueados hasta que el gestor de colas se reinicia o se reconecta al lote CICS, IMS o RRS.

## IBM MQ y IMS

IBM MQ proporciona dos componentes para interactuar con IMS, el adaptador IBM MQ - IMS y el puente IBM MQ - IMS. Estos componentes se denominan comúnmente adaptador IMS y puente IMS.

### Funcionamiento del adaptador IMS

Utilice este tema para comprender cómo funciona el adaptador IMS, que conecta IBM MQ con los sistemas IMS.

**Nota:** El adaptador IMS no incorpora ningún panel de operaciones ni de control.

Este tema contiene las siguientes secciones:

- [“Controlar las conexiones IMS” en la página 543](#)
- [“Conexión desde la región de control IMS” en la página 544](#)
- [“Visualización de unidades de recuperación pendientes” en la página 546](#)
- [“Control de las conexiones de regiones dependientes de IMS” en la página 548](#)
- [“Desconexión de IMS” en la página 550](#)
- [“Controlar el supervisor desencadenante de IMS” en la página 550](#)

### Controlar las conexiones IMS

En este tema se describen los mandatos de operador de IMS que controlan y supervisan las conexión a IBM MQ.

IMS proporciona los siguientes mandatos de operador para controlar y supervisar la conexión a IBM MQ:

#### **/CHANGE SUBSYS**

Suprime una unidad de recuperación pendiente de IMS.

#### **/DISPLAY OASN SUBSYS**

Muestra los elementos de recuperación pendientes.

## **/DISPLAY SUBSYS**

Muestra el estado de conexión y la actividad de hebras.

## **/START SUBSYS**

Conecta la región de control IMS a un gestor de colas.

## **/STOP SUBSYS**

Desconecta IMS de un gestor de colas.

## **/TRACE**

Controla el rastreo de IMS.

Para obtener más información sobre estos mandatos, consulte el manual *IMS/ESA Operator's Reference* correspondiente al nivel de IMS que está utilizando.

Las respuestas de los mandatos IMS se envían al terminal desde el que se emitió el mandato. La autorización para emitir mandatos IMS se basa en la seguridad de IMS.

## **z/OS Conexión desde la región de control IMS**

En este tema se explican los mecanismos disponibles para conectarse desde IMS a IBM MQ.

IMS establece una conexión desde su región de control a cada gestor de colas que utiliza IMS. IMS debe estar habilitado para establecer la conexión de una de estas maneras:

- Automáticamente durante:
  - Una inicialización de arranque en frío.
  - Un inicio en caliente de IMS, si la conexión de IBM MQ estaba activa cuando se concluyó IMS.
- En respuesta al mandato IMS:

```
/START SUBSYS sysid
```

donde *sysid* es el nombre del gestor de colas.

El mandato se puede emitir independientemente de si el gestor de colas está activo o no.

La conexión no se establece hasta que se realiza la primera llamada API de MQ al gestor de colas. Hasta ese momento, el mandato IMS /DIS SUBSYS muestra el estado como 'NO CONN'.

El orden en que se inicia IMS y el gestor de colas no es significativo.

IMS no puede volver a habilitar la conexión con el gestor de colas automáticamente si el gestor de colas se detiene con un mandato STOP QMGR, el mandato IMS /STOP SUBSYS o una finalización anómala. Por lo tanto, debe establecer la conexión mediante el mandato IMS /START SUBSYS.

Si se ve un mandato IMS en el registro de la consola del gestor de colas similar al siguiente:

```
MODIFY IMS*,SS*
```

Compruebe el registro maestro de IMS y asegúrese de que IBM MQ tenga autorización RACF para emitir mandatos MODIFY de IMS Adapter.

## **Inicializar el adaptador y conectar con el gestor de colas**

El adaptador es un conjunto de módulos que se cargan en las regiones de control y dependientes de IMS, utilizando el Recurso de conexión a subsistema externo de IMS.

Este procedimiento inicializa el adaptador y conecta con el gestor de colas:



1. Lea el miembro de subsistema (SSM) de IMS.PROCLIB. El SSM elegido es un parámetro EXEC de IMS. Hay una entrada en el miembro para cada gestor de colas al que IMS se puede conectar. Cada entrada contiene información de control sobre un adaptador de IBM MQ.

2. Cargue el adaptador IMS.

**Nota:** IMS carga una copia de los módulos de adaptador para cada instancia de IBM MQ definida en el miembro SSM.

3. Adjunte la tarea de subsistema externo para IBM MQ.

4. Ejecute el adaptador con el parámetro EXEC CTL ((IMSID) como el nombre de conexión.

El proceso es el mismo si la conexión es parte de la inicialización o resultado del mandato de IMS /START SUBSYS.

Si el gestor de colas está activo cuando IMS intenta establecer la conexión, se envían los mensajes siguientes:

- a la consola de z/OS:

```
DFS3613I ESS TCB INITIALIZATION COMPLETE
```

- al terminal maestro de IMS:

```
CSQQ000I IMS/TM imsid connected to queue manager ssnm
```

Cuando IMS intenta establecer la conexión y *el gestor de colas no está activo*, se envían los siguientes mensajes al terminal maestro de IMS cada vez que una aplicación realiza una llamada MQI:

```
CSQQ001I IMS/TM imsid not connected to queue manager ssnm.  
Notify message accepted  
DFS3607I MQM1 SUBSYSTEM ID EXIT FAILURE, FC = 0286, RC = 08,  
JOBNAME = IMSEMPR1
```

Si recibe mensajes DFS3607I cuando inicia la conexión a IMS o en el inicio del sistema, esto indica que el gestor de colas no está disponible. Para impedir que se genere un gran número de mensajes, debe realizar una de las siguientes acciones:

1. Inicie el gestor de colas correspondiente.
2. Emita el mandato de IMS:

```
/STOP SUBSYS
```

para que IMS no espere conectarse al gestor de colas.

Si no realiza ninguna de estas acciones, se emite un mensaje DFS3607I y el mensaje asociado CSQQ001I cada vez que se planifica un trabajo en la región y cada vez que una aplicación realiza una solicitud de conexión al gestor de colas.

## Conexión de hebras

En una región MPP o IFP, IMS establece una conexión de hebra cuando se planifica el primer programa de aplicación en esa región, incluso si ese programa de aplicación no realiza una llamada de IBM MQ. En una

región BMP, la conexión de hebra se establece cuando la aplicación realiza su primera llamada IBM MQ (MQCONN o MQCONNX). Esta hebra se conserva mientras dura la región o hasta que la conexión se detiene.

Tanto para las regiones dirigidas por mensajes como para las no dirigidas por mensajes, el identificador de referencia cruzada de hebra de recuperación, *Thread-xref*, asociado a la hebra es:

```
PSTid + PSBname
```

donde:

### **PSTid**

Identificador de región de tabla de especificaciones de partición

### **PSBname**

Nombre de bloque de especificación de programa

Puede utilizar identificadores (ID) de conexión como identificadores exclusivos en mandatos de IBM MQ, en cuyo caso IBM MQ inserta automáticamente estos ID en cualquier mensaje de operador que genera.

## **Visualización de unidades de recuperación pendientes**

Puede visualizar unidades de recuperación pendientes e intentar recuperarlas.

Los pasos operativos utilizados para listar y recuperar las unidades de recuperación pendientes en este tema son sólo para casos relativamente simples. Si el gestor de colas finaliza de forma anómala mientras está conectado a IMS, IMS puede confirmar o restituir trabajo sin que IBM MQ se percate de ello. Cuando el gestor de colas se reinicia, ese trabajo se denomina *pendiente*. Debe tomarse una decisión sobre el estado del trabajo.

Para visualizar una lista de unidades de recuperación pendientes, emita el mandato:

```
+CSQ1 DISPLAY CONN(*) WHERE(UOWSTATE EQ UNRESOLVED)
```

IBM MQ responde con un mensaje como el siguiente:

```
CSQM201I +CSQ1 CSQMDRTC DIS CONN DETAILS
CONN(BC0F6125F5A30001)
EXTCONN(C3E2D8C3C3E2D8F140404040404040)
TYPE(CONN)
CONNOPTS(
MQCNO_STANDARD_BINDING
)
UOWLOGDA(2004-11-02)
UOWLOGTI(12.27.58)
UOWSTDA(2004-11-02)
UOWSTTI(12.27.58)
UOWSTATE(UNRESOLVED)
NID(CSQ1CHIN.BC0F5F1C86FC0766)
EXTURID(0000000000000001F000000007472616E5F6964547565204E6F762020...)
QMURID(000000026232)
URTYPE(XA)
USERID( )
APPLTAG(CSQ1CHIN)
ASID(0000)
APPLTYPE(CHINIT)
CHANNEL( )
CONNNAME( )
END CONN DETAILS
```

Para obtener una explicación de los atributos de este mensaje, consulte la descripción del mandato [DISPLAY CONN](#).

## Recuperar unidades de recuperación pendientes

Para recuperar las unidades de recuperación pendientes, emita este mandato:

```
+CSQ1 RESOLVE INDOUBT( connection-name ) ACTION(COMMIT|BACKOUT)
NID( net-node.number )
```

donde:

### ***connection-name***

El ID de sistema de IMS.

### **ACTION**

Indica si se debe confirmar (COMMIT) o restituir (BACKOUT) esta unidad de recuperación.

### ***net-node.number***

El nodo-red.número asociado.

Cuando haya emitido el mandato RESOLVE INDOUBT, aparecerá uno de los siguientes mensajes:

```
CSQV414I +CSQ1 THREAD network-id COMMIT SCHEDULED
CSQV415I +CSQ1 THREAD network-id BACKOUT SCHEDULED
```

## Resolver de entradas de recuperación residuales

En determinados momentos, IMS crea una lista de entradas de recuperación residuales (RREs). Las RRE son unidades de recuperación sobre las que IBM MQ puede tener dudas. Pueden surgir en varias situaciones:

- Si el gestor de colas no está activo, IMS tiene RREs que no se pueden resolver hasta que el gestor de colas esté activo. Estas RRE no son un problema.
- Si el gestor de colas está activo y conectado a IMS, y si IMS restituye el trabajo que IBM MQ ha confirmado, el adaptador IMS emite el mensaje CSQQ010E. Si los datos de los dos sistemas deben ser coherentes, hay un problema. Para obtener más información sobre cómo solucionar este problema, consulte [“Recuperar de unidades de recuperación IMS manualmente”](#) en la página 540.
- Si el gestor de colas está activo y conectado a IMS, puede que todavía haya entradas de recuperación residual (RRE), aunque ningún mensaje le haya informado de este problema. Una vez establecida la conexión de IBM MQ a IMS, puede emitir el siguiente mandato de IMS para averiguar si hay un problema:

```
/DISPLAY OASN SUBSYS sysid
```

Para depurar la RRE, emita uno de los siguientes mandatos de IMS:

```
/CHANGE SUBSYS sysid RESET
/CHANGE SUBSYS sysid RESET OASN nnnn
```

donde *nnnn* es el número de secuencia de aplicación de origen listado en respuesta al mandato +CSQ1 DISPLAY. Este es el número de planificación de la instancia del programa, que indica su lugar en la secuencia de invocaciones de ese programa desde el último arranque en frío de IMS. IMS no puede tener dos unidades de recuperación pendientes con el mismo número de planificación.

Estos mandatos restablecen el estado de IMS; no dan como resultado ninguna comunicación con IBM MQ.

## **Control de las conexiones de regiones dependientes de IMS**

Puede controlar, supervisar y, cuando sea necesario, terminar las conexiones entre IMS e IBM MQ.

El control de las conexiones de regiones dependientes de IMS incluye las siguientes actividades:

- [Conectarse desde regiones dependientes](#)
- [Opciones de error de región](#)
- [Supervisar la actividad en las conexiones](#)
- [Desconectarse desde regiones dependientes](#)

### **Conectarse desde regiones dependientes**

El adaptador IMS que se utiliza en la región de control también se carga en las regiones dependientes. Se establece una conexión desde cada región dependiente a IBM MQ. Esta conexión se utiliza para coordinar la confirmación de IBM MQ y el trabajo de IMS. Para inicializar y establecer la conexión, IMS realiza lo siguiente:

1. Lee el miembro de subsistema (SSM) desde IMS.PROCLIB.

Se puede especificar un miembro de subsistema en el parámetro EXEC de la región dependiente. Si no se especifica, se utiliza el SSM de la región de control. Si no es probable que la región se conecte a IBM MQ, para evitar cargar el adaptador, especifique un miembro sin entradas.

2. Carga el adaptador de IBM MQ.

Para un programa de mensajes por lotes, la carga no se realiza hasta que la aplicación emite su primer mandato de mensajería. En ese momento, IMS intenta establecer la conexión.

Para una región de programa de proceso de mensajes o una región de vía de acceso rápida de IMS, el intento se lleva a cabo cuando se inicializa la región.

### **Opciones de error de región**

Si el gestor de colas no está activo, o si no hay recursos disponibles cuando se envía el primer mandato de mensajería desde los programas de aplicación, la acción realizada depende de la opción de error especificada en la entrada SSM. Las opciones son:

#### **R**

El código de retorno apropiado se envía a la aplicación.

#### **Q**

La aplicación finaliza de forma anómala con el código de terminación anómala U3051. El mensaje de entrada se vuelve a poner en cola.

#### **A**

La aplicación finaliza de forma anómala con el código de terminación anómala U3047. El mensaje de entrada se descarta.

### **Supervisar la actividad en las conexiones**

Se establece una hebra desde una región dependiente cuando una aplicación realiza su primera solicitud de IBM MQ satisfactoria. Puede visualizar información sobre conexiones y las aplicaciones que actualmente las utilizan emitiendo el siguiente mandato desde IBM MQ.

```
+CSQ1 DISPLAY CONN(*) ALL
```

El mandato genera un mensaje como el siguiente:

```
CONN(BC45A794C4290001)
EXTCONN(C3E2D8C3C3E2D8F140404040404040)
TYPE(CONN)
CONNOPTS(
MQCNO_STANDARD_BINDING
)
UOWLOGDA(2004-12-15)
UOWLOGTI(16.39.43)
UOWSTDA(2004-12-15)
UOWSTTI(16.39.43)
UOWSTATE(ACTIVE)
NID( )
EXTURID(
0000052900000000
)
QMURID(00000354B76E)
URTYPE(IMS)
USERID(STCPI)
APPLTAG(IM8F)
ASID(0049)
APPLTYPE(IMS)
PSTID(0004)
PSBNAME(GP01MPP)
```

Para la región de control, *thread-xref* es el valor especial CONTROL. Para regiones dependientes, es el PSTid concatenado con el PSBname. *auth-id* es el campo de usuario de la tarjeta de trabajos o el ID de la tabla de procedimientos iniciados de z/OS.

Para obtener una explicación de la lista visualizada, consulte la descripción del mensaje CSQV402I en la documentación [IBM MQ for z/OS mensajes, finalización, y códigos de razón](#).

IMS proporciona un mandato de visualización para supervisar la conexión con IBM MQ. El mandato muestra qué programa está activo en cada conexión de región dependiente, el nombre de usuario de LTERM y el estado de conexión de la región de control. El mandato es:

```
/DISPLAY SUBSYS name
```

El estado de la conexión entre IMS e IBM MQ se muestra como uno de los siguientes:

```
CONNECTED
NOT CONNECTED
CONNECT IN PROGRESS
STOPPED
STOP IN PROGRESS
INVALID SUBSYSTEM NAME= name
SUBSYSTEM name NOT DEFINED BUT RECOVERY OUTSTANDING
```

El estado de hebra de cada región dependiente es uno de los siguientes:

```
CONN
CONN, ACTIVE (includes LTERM of user)
```

## Desconectarse de regiones dependientes

Para cambiar valores en el miembro SSM de IMS.PROCLIB, debe desconectar una región dependiente. Para ello, debe:

1. Emita el mandato de IMS:

```
/STOP REGION
```

2. Actualizar el miembro SSM.
3. Emita el mandato de IMS:

```
/START REGION
```

## **Desconexión de IMS**

La conexión se finaliza cuando IMS o el gestor de colas termina. De forma alternativa, el operador de terminal maestro de IMS puede interrumpir explícitamente la conexión.

Para terminar la conexión entre IMS e IBM MQ, utilice el siguiente mandato de IMS:

```
/STOP SUBSYS sysid
```

El mandato envía el siguiente mensaje al terminal que lo emitió, normalmente el operador de terminal maestro (MTO):

```
DFS058I STOP COMMAND IN PROGRESS
```

El mandato IMS:

```
/START SUBSYS sysid
```

es necesario para restablecer la conexión.

**Nota:** El mandato de IMS /STOP SUBSYS no se completará si hay un supervisor desencadenante de IMS en ejecución.

## **Controlar el supervisor desencadenante de IMS**

Puede utilizar la transacción CSQQTRMN para detener e iniciar el supervisor desencadenante de IMS.

El supervisor desencadenante de IMS (la transacción CSQQTRMN) se describe en [Configuración del supervisor desencadenante de IMS](#).

Para controlar el supervisor desencadenante de IMS, consulte:

- [Inicio de CSQQTRMN](#)
- [Detención de CSQQTRMN](#)

## Inicio de CSQQTRMN

1. Inicie un BMP orientado a lotes que ejecute el programa CSQQTRMN para cada cola de inicio que desee supervisar.
2. Modifique el JCL de proceso por lotes para añadir un DDname de CSQQUT1 que apunte a un conjunto de datos que contenga la siguiente información:

```
QMGRNAME=q_manager_name    Comment: queue manager name
INITQUEUEUENAME=init_q_name  Comment: initiation queue name
LTERM=lterm                  Comment: LTERM to remove error messages
CONSOLEMESSAGES=YES         Comment: Send error messages to console
```

donde:

q_manager_name	El nombre del gestor de colas (si está en blanco, se presupone el valor predeterminado designado en CSQQDEFV)
init_q_name	El nombre de la cola de inicio que se va a supervisar
lterm	El nombre del terminal lógico (LTERM) de IMS para el destino de los mensajes de error (si está en blanco, el valor predeterminado es MASTER).
CONSOLEMESSAGES= YES	Solicita que los mensajes que se envíen al terminal lógico (LTERM) de IMS designado también se envíen a la consola de z/OS. Si este parámetro se omite o se escribe incorrectamente, el valor predeterminado es no enviar mensajes a la consola (NOT).

3. Añada un nombre DD de CSQQUT2 si desea un informe impreso del proceso de la entrada CSQQUT1.

### Nota:

1. El conjunto de datos CSQQUT1 se define con LRECL=80. Otra información de DCB se obtiene del conjunto de datos. El DCB para el conjunto de datos CSQQUT2 es RECFM=VBA y LRECL=125.
2. Puede poner una sola palabra clave en cada registro. El valor de la palabra clave está delimitado por el primer espacio en blanco después de la palabra clave; esto significa que puede incluir comentarios. Un asterisco en la columna 1 significa que todo el registro de entrada es un comentario.
3. Si escribe incorrectamente cualquiera de las palabras clave QMGRNAME o LTERM, CSQQTRMN utiliza el valor predeterminado para esa palabra clave.
4. Asegúrese de que el subsistema se ha iniciado en IMS (mediante el mandato /START SUBSYS) antes de someter el trabajo BMP de supervisor desencadenante. Si no se ha iniciado, el trabajo de supervisor desencadenante termina con el código de terminación anómala U3042.

## Detención de CSQQTRMN

Una vez iniciado, CSQQTRMN se ejecuta hasta que la conexión entre IBM MQ e IMS se interrumpe debido a uno de los sucesos siguientes:

- el gestor de colas finaliza
- IMS finaliza

o se especifica un mandato de z/OS STOP **jobname**.

## **Controlar el puente IMS**

En este tema se describen los mandatos IMS que puede utilizar para controlar el puente IMS.

No hay mandatos de IBM MQ para controlar el puente IBM MQ-IMS. No obstante, puede impedir que se entreguen mensajes a IMS de las siguientes maneras:

- Para colas no compartidas, mediante el mandato ALTER QLOCAL(xxx) GET(DISABLED) para todas las colas puente.
- Para colas agrupadas en clúster, mediante el mandato SUSPEND QMGR CLUSTER(xxx). Esto sólo es efectivo cuando otro gestor de colas también está alojando la cola puente agrupada en clúster.
- Para colas en clúster, mediante el mandato SUSPEND QMGR FACILITY(IMSBRIDGE). No se envían más mensajes a IMS, pero se reciben de IMS las respuestas para las transacciones pendientes.

Para empezar de nuevo a enviar mensajes a IMS, emita el mandato RESUME QMGR FACILITY(IMSBRIDGE).

También puede utilizar el mandato MQSC DISPLAY SYSTEM para ver si el puente está suspendido.

Consulte [Mandatos MQSC](#) para obtener información detallada acerca de estos mandatos.

Para más información, consulte:

- [“Iniciar y detener el puente IMS” en la página 552](#)
- [“Controlar las conexiones IMS” en la página 552](#)
- [Controlar las colas puente](#)
- [“Resincronizar el puente IMS” en la página 554](#)
- [Trabajar con nombres de tpipe](#)
- [Suprimir mensajes de IMS](#)
- [Suprimir tpipes](#)
- [“Caducidad de transacciones IMS” en la página 555](#)

## **Iniciar y detener el puente IMS**

Inicie el puente de IBM MQ iniciando OTMA. Utilice el mandato IMS:

```
/START OTMA
```

o inícielo automáticamente especificando OTMA=YES en los parámetros del sistema IMS. Si OTMA ya está iniciado, el puente se inicia automáticamente una vez se ha completado el inicio del gestor de colas. Se genera un mensaje de suceso de IBM MQ cuando se inicia OTMA.

Utilice el mandato IMS:

```
/STOP OTMA
```

para detener la comunicación con OTMA. Cuando se emite este mandato, se genera un mensaje de suceso de IBM MQ.

## **Controlar las conexiones IMS**

IMS proporciona estos mandatos de operador para controlar y supervisar la conexión a IBM MQ:



**/DEQUEUE TMEMBER *tmember* TPIPE *tpipe***

Elimina mensajes de un Tpipe. Especifique PURGE para eliminar todos los mensajes o PURGE1 para eliminar sólo el primer mensaje.

**/DISPLAY OTMA**

Muestra información de resumen sobre el servidor y los clientes OTMA, y el estado del cliente.

**/DISPLAY TMEMBER *nombre***

Muestra información sobre un cliente OTMA.

**/DISPLAY TRACE TMEMBER *nombre***

Muestra información sobre lo que se está rastreando.

**/SECURE OTMA**

Establece opciones de seguridad.

**/START OTMA**

Habilita las comunicaciones a través de OTMA.

**/START TMEMBER *tmember* TPIPE *tpipe***

Inicia el Tpipe especificado.

**/STOP OTMA**

Detiene las comunicaciones a través de OTMA.

**/STOP TMEMBER *tmember* TPIPE *tpipe***

Detiene el Tpipe especificado.

**/TRACE**

Controla el rastreo de IMS.

Para obtener más información sobre estos mandatos, consulte el manual *IMS/ESA Operators Reference* correspondiente al nivel de IMS que está utilizando.

Las respuestas de los mandatos IMS se envían al terminal desde el que se emitió el mandato. La autorización para emitir mandatos IMS se basa en la seguridad de IMS.

## Controlar las colas puente

Para dejar de comunicarse con el gestor de colas con el nombre de miembro XCF *tmember* a través del puente, emita el siguiente mandato IMS:

```
/STOP TMEMBER tmember TPIPE ALL
```

Para reanudar la comunicación, emita el siguiente mandato IMS:

```
/START TMEMBER tmember TPIPE ALL
```

Los Tpipes para una cola se pueden visualizar mediante el mandato MQ DISPLAY QUEUE.

Para detener la comunicación con el gestor de colas en un Tpipe individual, emita el siguiente mandato IMS:

```
/STOP TMEMBER tmember TPIPE tpipe
```

Se crean uno o dos Tpipes para cada cola puente activa, por lo que emitir este mandato detiene la comunicación con la cola de IBM MQ. Para reanudar la comunicación, utilice el siguiente mandato IMS:

```
/START TMEMBER tmember TPIPE tpipe
```

O bien, puede modificar los atributos de la cola de IBM MQ para inhibirla para obtener.

## Resincronizar el puente IMS

El puente IMS se reinicia automáticamente cada vez que se reinicia el gestor de colas, IMS u OTMA.

La primera tarea realizada por el puente IMS es resincronizarse con IMS. Esto implica que IBM MQ e IMS comprueben los números de secuencia en cada Tpipe sincronizado. Se utiliza un Tpipe sincronizado cuando se envían mensajes persistentes a IMS desde una cola puente IBM MQ - IMS utilizando la modalidad de confirmación cero (confirmar-luego-enviar).

Si el puente no se puede resincronizar con IMS, se devuelve el código de detección de IMS en el mensaje CSQ2023E y se detiene la conexión a OTMA. Si el puente no se puede resincronizar con un Tpipe de IMS individual, se devuelve el código de detección IMS en el mensaje CSQ2025E y se detiene el Tpipe. Si un Tpipe se ha arrancado en frío, los números de secuencia recuperables se restablecen automáticamente a 1.

Si el puente descubre números de secuencia no coincidentes al resincronizar con un Tpipe, se emite el mensaje CSQ2020E. Utilice el mandato de IBM MQ RESET TPIPE para iniciar la resincronización con el Tpipe de IMS. Debe proporcionar el nombre de grupo y de miembro XCF y el nombre del Tpipe; esta información se facilita en el mensaje.

También puede especificar:

- Un nuevo número de secuencia recuperable para establecerlo en el Tpipe para los mensajes enviados por IBM MQ y para establecerlo como el número de secuencia de recepción del asociado. Si no especifica esto, el número de secuencia de recepción del asociado se establece en el número de secuencia de envío actual de IBM MQ.
- Un nuevo número de secuencia recuperable para establecerlo en el Tpipe para los mensajes recibidos por IBM MQ y para establecerlo como el número de secuencia de envío del asociado. Si no especifica esto, el número de secuencia de envío del asociado se establece en el número de secuencia de recepción actual de IBM MQ.

Si hay una unidad de recuperación sin resolver asociada al Tpipe, esto también se notifica en el mensaje. Utilice el mandato de IBM MQ RESET TPIPE para especificar si desea confirmar la unidad de recuperación o restituirla. Si confirma la unidad de recuperación, el lote de mensajes ya se ha enviado a IMS, y se suprime de la cola puente. Si restituye la unidad de recuperación, los mensajes se devuelven a la cola puente, para enviarlos posteriormente a IMS.

Los Tpipes de modalidad de confirmación 1 (enviar-luego-confirmar) no se sincronizan.

### Consideraciones para las transacciones de Modalidad de confirmación 1

En IMS, la transacciones de modalidad de confirmación 1 (CM1) envían sus respuestas de salida antes del punto de sincronización.

Es posible que una transacción CM1 no pueda enviar su respuesta, por ejemplo porque:

- El Tpipe en el que se va a enviar la respuesta se ha detenido
- OTMA se ha detenido
- El cliente OTMA (es decir, el gestor de colas) ha desaparecido
- La cola de respuesta y la cola de mensajes no entregados no están disponibles

Por estas razones, la aplicación IMS que envía el mensaje seudotermina de forma anómala con el código U0119. La transacción y el programa IMS no se detienen en este caso.

Estas razones a menudo impiden que se envíen mensajes a IMS, así como que se entreguen respuestas desde IMS. Una terminación anómala U0119 se puede producir si:

- Se detiene el Tpipe, OTMA o el gestor de colas mientras el mensaje está en IMS
- IMS responde en un Tpipe distinto al del mensaje entrante, y ese Tpipe se ha detenido
- IMS responde a un cliente OTMA diferente, y ese cliente no está disponible.

Cada vez que se produce una terminación anómala U0119, tanto el mensaje entrante a IMS como los mensajes de respuesta a IBM MQ se pierden. Si la salida de una transición CMO no se puede entregar por alguna de las razones anteriormente mencionadas, se pone en cola en el Tpipe dentro de IMS.

## Trabajar con nombres de tpipe

Muchos de los mandatos que se utilizan para controlar el puente IBM MQ - IMS requieren el nombre de *tpipe*. Lea este tema para saber cómo puede encontrar más información sobre el nombre de *tpipe*.

Necesita nombres de *tpipe* para muchos de los mandatos que controlan el puente IBM MQ - IMS. Puede obtener los nombres de *tpipe* del mandato DISPLAY QUEUE, y tenga en cuenta los puntos siguientes:

- los nombres de *tpipe* se asignan cuando se define una cola local
- a una cola local se le asignan dos nombres de *tpipe*, uno para sincronización y uno para no sincronización
- Los nombres de *tpipe* no serán conocidos en IMS hasta que no se haya establecido alguna comunicación entre IMS e IBM MQ específica para esa cola local.
- Para que un *tpipe* pueda ser utilizado por el puente IBM MQ - IMS, su cola asociada debe estar asignada a una clase de almacenamiento que tenga el grupo XCF correcto y cuyos campos de nombre de campo se hayan completado.

## Suprimir mensajes de IMS

Un mensaje destinado a IBM MQ a través del puente IMS se puede suprimir si el Tmember/Tpipe se ha detenido. Para suprimir un mensaje para el gestor de colas con el nombre de miembro XCF *tmember*, emita el siguiente mandato IMS:

```
/DEQUEUE TMEMBER tmember TPIPE tpipe PURGE1
```

Para suprimir todos los mensajes en el Tpipe, emita el siguiente mandato IMS:

```
/DEQUEUE TMEMBER tmember TPIPE tpipe PURGE
```

## Suprimir tpipes (conductos de transacción)

No puede suprimir tpipes IMS usted mismo. Los tpipes son suprimidos por IMS en las siguientes ocasiones:

- Los tpipes sincronizados se suprimen cuando se realiza un arranque en frío de IMS.
- Los tpipes no sincronizados se suprimen cuando se reinicia IMS.

## Caducidad de transacciones IMS

Hay un tiempo de caducidad asociado a una transacción; cualquier mensaje de IBM MQ puede tener asociado un tiempo de caducidad. El intervalo de caducidad se pasa de la aplicación a IBM MQ utilizando

el campo MQMD.Expiry. El tiempo es la duración de un mensaje antes de que caduque, expresado como un valor en décimas de segundo. El resultado de un intento de realizar la operación MQGET de un mensaje, después que haya caducado es que el mensaje se elimina de la cola y se lleva a cabo el proceso de caducidad. El tiempo de caducidad disminuye que un mensaje fluye entre los gestores de colas de una red de IBM MQ. Cuando un mensaje IMS se pasa a través del puente IMS a OTMA, el tiempo de caducidad de mensaje restante se pasa a OTMA como el tiempo de caducidad de transacción.

Si una transacción tiene un tiempo de caducidad específico, OTMA caduca las transacciones de entrada en tres lugares diferentes de IMS:

- en un mensaje de entrada que se recibe de XCF
- en el tiempo de transferencia de mensajes en cola
- en el tiempo GU de la aplicación

Después de la hora GU no se realiza ninguna caducidad.

La variable EXPRTIME se puede proporcionar mediante:

- la definición de transacción IMS
- la cabecera de mensaje OTMA de IMS
- la salida de usuario DFSINSX0 de IMS
- los mandatos IMS CREATE o UPDATE TRAN

IMS indica que ha caducado una transacción terminando de forma anómala una transacción con 0243 y emitiendo un mensaje. El mensaje emitido es DFS555I en el entorno de colas no compartidas o bien DFS2224I en el entorno de colas compartidas.

## Operación de Advanced Message Security en z/OS

El espacio de direcciones de Advanced Message Security acepta mandatos mediante el mandato MODIFY de z/OS.

Para especificar mandatos para el espacio de direcciones de Advanced Message Security (AMS), utilice el mandato MODIFY de z/OS.

Por ejemplo,

```
F qmgrAMSM, cmd
```

donde *qmgr* es el prefijo del nombre de la tarea iniciada.

Tabla 34 en la página 556 describe los mandatos MODIFY que están aceptados:

<i>Tabla 34. Mandatos MODIFY del espacio de direcciones de Advanced Message Security</i>		
<b>Mandato</b>	<b>Opción</b>	<b>Descripción</b>
DISPLAY		Visualizar información de versión
REFRESH	KEYRING política ALL	Renovar los certificados del conjunto de claves, las políticas de seguridad o ambas cosas.
SMFAUDIT	SUCCESS FAILURE ALL	Establezca si la auditoría de SMF es necesaria cuando AMS protege o anula la protección de mensajes correctamente, cuando AMS no puede proteger o anular la protección de mensajes, o en ambos casos.

Tabla 34. Mandatos MODIFY del espacio de direcciones de Advanced Message Security (continuación)

Mandato	Opción	Descripción
SMFTYPE	0 - 255	Establezca el tipo de registro SMF que se va a generar cuando AMS protege o anula la protección de mensajes. Para inhabilitar la auditoría SMF especifique un tipo de registro de 0.

**Nota:** Para especificar una opción debe separarla mediante una coma. Por ejemplo:

```
F qmgrAMSM, REFRESH KEYRING
F qmgrAMSM, SMFAUDIT ALL
F qmgrAMSM, SMFTYPE 180
```

## Mandato REFRESH

Los cambios que se han hecho efectivos emitiendo el mandato **REFRESH** se aplican a aplicaciones que emiten MQOPEN después de que el mandato **REFRESH** se haya completado. Las aplicaciones existentes que tienen una cola abierta siguen utilizando las opciones desde cuando la aplicación abrió la cola. Para utilizar los nuevos valores, la aplicación debe cerrar y volver a abrir la cola.

## Iniciar y detener AMS

No tendrá que especificar un mandato para iniciar o detener el espacio de direcciones de Advanced Message Security. El espacio de direcciones de AMS se inicia automáticamente cuando el gestor de colas se inicia, si AMS se ha habilitado con el parámetro **SPLCAP** de CSQ6SYSP, y se detiene cuando el gestor de colas se detiene.

## Proveedor de servicios IBM MQ for z/OS para IBM z/OS Connect EE

El proveedor de servicios IBM MQ for z/OS para IBM z/OS Connect EE (MQ Service Provider) procesa solicitudes que llegan a través de IBM z/OS Connect EE. MQ Service Provider permite que las aplicaciones que utilizan REST interactúen con activos de z/OS expuestos mediante el uso de colas y temas de IBM MQ for z/OS. Puede lograrlo sin tener que preocuparse de la codificación necesaria para utilizar la mensajería asíncrona.

**Importante:** z/OS Connect EE Versión 3.0.21.0 y posterior, incluye una versión mejorada del MQ Service Provider que admite archivos de archivado de servicio. Debe migrar a esa versión de z/OS Connect EE y utilizar el MQ Service Provider incorporado, en lugar de utilizar el proveedor de servicios que se suministra con el producto IBM MQ for z/OS.

En la información de z/OS Connect EE en IBM Documentation, los escenarios de inicio rápido para MQ Service Provider en z/OS Connect EE se proporcionan en Casos de ejemplo de inicio rápido, y la información de referencia detallada se proporciona en [Utilización del proveedor de servicios de IBM MQ](#).

Esta sección consta de la siguiente información:

### Información relacionada

Vídeo: [IBM MQ Service Provider for z/OS Connect \(YouTube\)](#)

## Proveedor de servicios IBM MQ for z/OS para IBM z/OS Connect EE - Descripción General

Una descripción general del proveedor de servicios IBM MQ for z/OS para IBM z/OS Connect EE, que explica los principios utilizados y los verbos que utiliza el proveedor de servicios.

**Importante:** z/OS Connect EE versión 3.0.21.0 y posteriores, envía una versión mejorada de MQ Service Provider que da soporte a los archivos de archivado de servicio. Debe migrar a esa versión de z/OS Connect EE y utilizar el MQ Service Provider incorporado, en lugar de utilizar el proveedor de servicios que se entrega con el producto IBM MQ for z/OS.

En la información de z/OS Connect EE en IBM Documentation, se proporcionan escenarios de inicio rápido para MQ Service Provider en z/OS Connect EE en [Casos de ejemplo de inicio rápido](#), y se proporciona información de referencia detallada en [Utilización del proveedor de servicios de IBM MQ](#).

El MQ Service Provider admite IBM z/OS Connect EE:

### **IBM z/OS Connect Enterprise Edition (z/OS Connect EE)**

Se trata de un producto individual cargable basado en WebSphere Liberty (WLP), que proporciona herramientas para generar diversas API RESTful, el editor de API y la integración con IBM API Connect.

Para obtener más información, consulte [IBM z/OS Connect EE](#).

**Importante:** El MQ Service Provider solo admite IBM z/OS Connect EE V2.0.3.0 (APAR [PI66869](#)) o posterior.

Todas las prestaciones de IBM z/OS Connect EE, como el editor de API, están soportadas con el MQ Service Provider.

Los activos de z/OS expuestos por z/OS Connect no necesitan conocer REST o JSON, porque z/OS Connect correlaciona las invocaciones REST con las llamadas locales y también proporciona transformaciones para convertirlas entre JSON y estructuras de datos locales, como libros de copias COBOL.

MQ Service Provider está disponible como una característica WLP independiente que puede instalarse en la instalación de z/OS Connect relevante. Tenga en cuenta que la mayoría del trabajo para configurar MQ Service Provider se basa en la configuración, de modo que la aplicación REST no necesita conocer IBM MQ. Sin embargo, se proporciona acceso a elementos como los campos MQMD para aplicaciones más avanzadas.

MQ Service Provider expone colas y temas de IBM MQ como [“Servicios” en la página 559](#).

## **Versiones soportadas de IBM MQ**

MQ Service Provider está soportado con IBM MQ for z/OS 8.0 y posteriores.

Las versiones de IBM MQ anteriores a IBM MQ 9.1.0 no se entregan con MQ Service Provider. En su lugar, consulte [“Obtención de los MQ Service Provider” en la página 562](#) para obtener información sobre cómo puede obtener MQ Service Provider.

La información descrita en los temas de esta sección de z/OS Connect se aplica igualmente a todas las versiones soportadas de IBM MQ.

## **Modalidades de conexión**

**V 9.1.1**

MQ Service Provider utiliza IBM MQ classes for JMS para conectarse a un gestor de colas.

IBM MQ classes for JMS proporciona dos modalidades de conexión:

### **Modalidad de cliente**

En la modalidad de cliente, las conexiones al gestor de colas se realizan a través de TCP/IP.

### **Modalidad de enlaces**

En la modalidad de enlaces, las conexiones se realizan directamente al gestor de colas utilizando la interfaz nativa Java (JNI).

Si desea más información sobre las modalidades de conexión, consulte: [Modalidades de conexión para IBM MQ classes for JMS](#).

MQ Service Provider en IBM MQ 9.1.0 admite conexiones a los gestores de colas que solo se ejecutan en las mismas LPAR que el servidor IBM z/OS Connect EE y las conexiones se deben realizar mediante la modalidad de enlaces.

Desde IBM MQ 9.1.1, el proveedor de servicios se ha ampliado para dar soporte también a las conexiones de modalidad cliente con gestores de colas IBM MQ for z/OS locales o remotos. Esto permite una mayor flexibilidad en la configuración del despliegue de z/OS Connect EE.

Para el MQ Service Provider, la modalidad de conexión se configura utilizando la propiedad **transportType** en la sección `properties.wmqJms` de la fábrica de conexiones en el archivo `server.xml` del servidor IBM z/OS Connect EE:

- Un **transportType** de BINDINGS permite conexiones en modalidad de enlaces.
- Un **transportType** de CLIENT permite conexiones en modalidad cliente.

Si son necesarias conexiones en modalidad cliente, también debe establecer las propiedades **channel**, **hostName** y **port** y, probablemente, también los distintos atributos SSL, si se está utilizando TLS.

Consulte [properties.wmqJMS](#) si desea ver detalles de estas propiedades.

## Otras consideraciones

MQ Service Provider utiliza un:

- HTTP GET para una llamada MQGET no destructiva
- HTTP DELETE para una llamada MQGET destructiva
- HTTP POST para una llamada MQPUT

El MQ Service Provider utiliza cabeceras HTTP como una forma de especificar varias cosas como, por ejemplo, valores MQMD.

Si desea más información sobre las llamadas HTTP, consulte [“Verbos que MQ Service Provider soporta” en la página 561](#).

Tenga en cuenta que los únicos datos que se representa como datos JSON son la carga útil del mensaje. Esto significa que las aplicaciones no se exponen innecesariamente a los conceptos de IBM MQ si lo único que necesitan son los datos de mensaje.

MQ Service Provider utiliza el soporte de adaptador de recursos de IBM MQ en WLP y, por consiguiente, se basa en JMS.

## Servicios

MQ Service Provider expone las colas y los temas de IBM MQ, y las aplicaciones que están tras ellos, como servicios. Hay dos tipos de servicios: unidireccional y bidireccional que se describen en esta sección.

### Servicio unidireccional

Un servicio unidireccional puede utilizarse para proporcionar una API RESTful sobre una sola cola o tema de IBM MQ. Los clientes RESTful pueden emitir un HTTP POST con una carga útil JSON en un servicio unidireccional y el servicio toma la carga útil y envía un mensaje, con la carga útil como el cuerpo del mensaje, la cola o tema de destino.

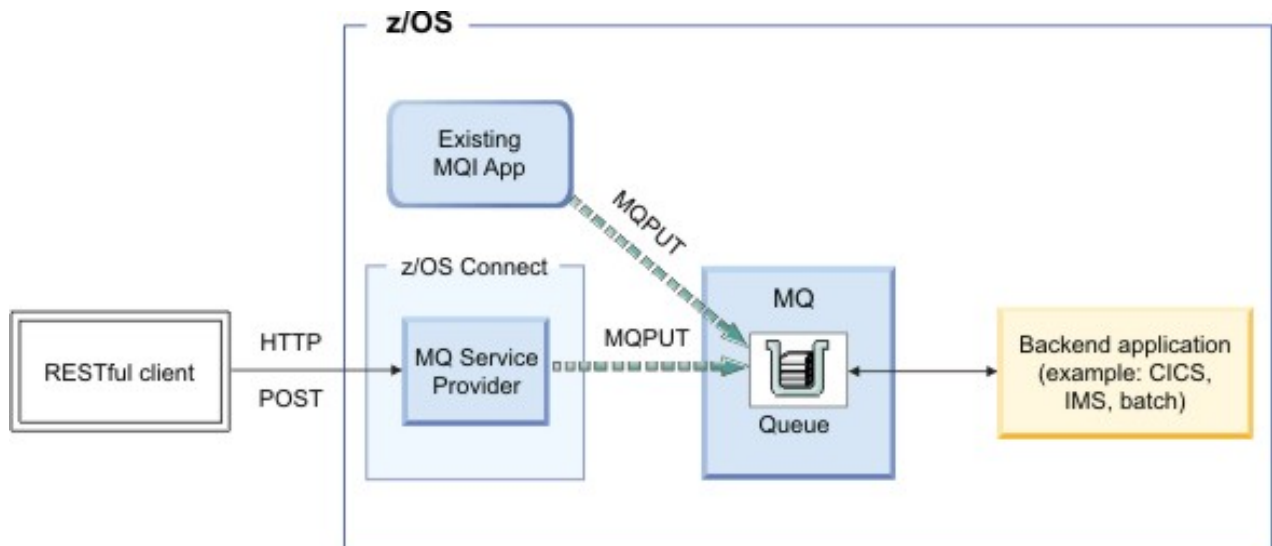


Figura 49. Servicio unidireccional

Los servicios unidireccionales también permiten que se emitan solicitudes HTTP DELETE y HTTP GET en las colas de IBM MQ.

Un HTTP DELETE produce una obtención destructiva de un mensaje disponible en la cola. Un HTTP GET produce un examen del primer mensaje disponible en la cola.

**Nota:** Dos llamadas HTTP GET devuelven el mismo mensaje, a menos que se haya producido alguna otra acción, por ejemplo, un HTTP DELETE o la caducidad de mensajes, para eliminar el mensaje de la cola.

El cuerpo del mensaje se devuelve al cliente RESTful en forma de JSON. Si el cuerpo del mensaje aún no está en formato JSON, por ejemplo un libro de copias COBOL, z/OS Connect puede configurarse para convertirlo a JSON, utilizando una transformación de datos.

## Servicio bidireccional

Un servicio bidireccional permite que un cliente RESTful realice la mensajería de solicitud-respuesta para un par de colas.

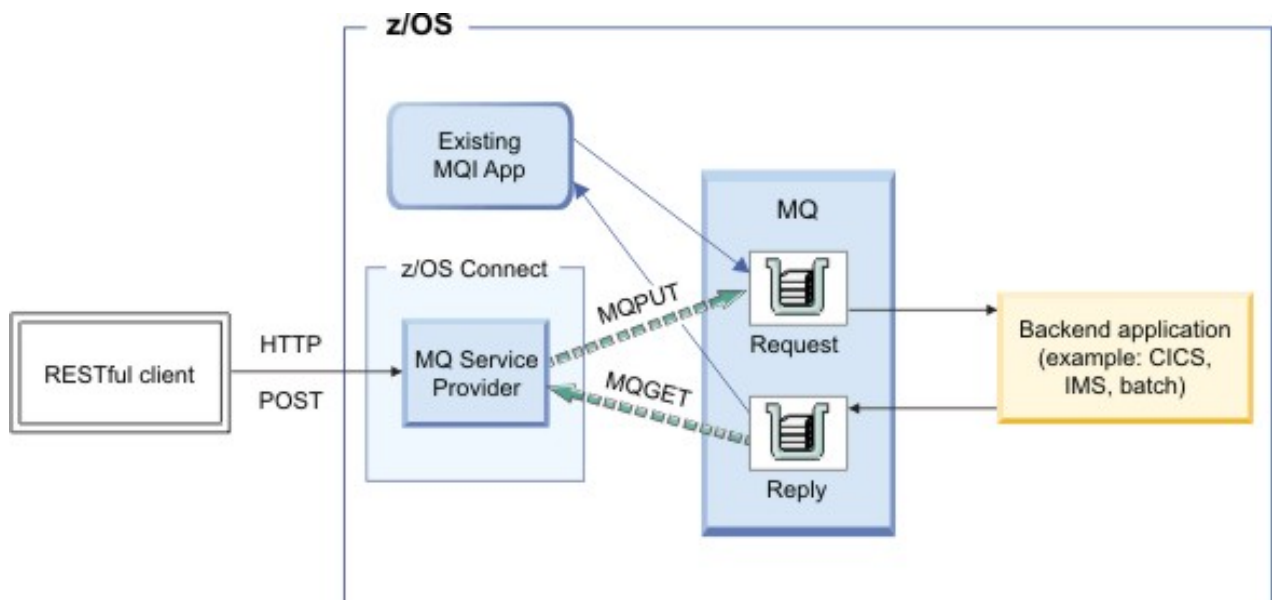


Figura 50. Servicio bidireccional



El cliente emite una solicitud HTTP POST especificando una carga útil JSON. El servicio toma la carga útil, opcionalmente la convierte a un formato diferente, como un libro de copias COBOL y lo envía como un mensaje a una cola de solicitud.

Una aplicación de fondo consume el mensaje, lo procesa y genera una respuesta que se coloca en una cola de respuesta. El servicio localiza este mensaje, toma la carga útil, opcionalmente la convierte a JSON y la devuelve como el cuerpo de respuesta del HTTP POST.

## **Verbos que MQ Service Provider soporta**

MQ Service Provider admite los verbos HTTP GET, HTTP DELETE y HTTP POST. El uso de estos verbos permite la interacción básica con IBM MQ for z/OS y la composición de varios de estos verbos permite exponer funciones más complicadas.

Al realizar un HTTP GET o HTTP DELETE, MQ Service Provider ignora cualquier cuerpo proporcionado en la solicitud.

Si HTTP GET o HTTP DELETE es satisfactoria:

- La carga útil del mensaje se devuelve en el cuerpo de respuesta HTTP y el código de estado HTTP es 200.
- Un mensaje sin carga útil se devuelve sin cuerpo HTTP. Si no hay ningún mensaje disponible, no se devuelve ningún cuerpo HTTP y el código de estado HTTP es 204 (Sin contenido).



**Atención:** IBM z/OS Connect EE espera que las cargas útiles JSON estén siempre en formato de objeto; es decir, la carga útil mínima válida es `{}`.

Al ejecutar una POST HTTP, los únicos parámetros esperados son la carga útil de mensaje, que estará en el cuerpo HTTP, y los diversos elementos descritos en [Elemento mqzOSConnectService](#).

Si la POST es satisfactoria, un:

### **Servicio unidireccional**

Devuelve el código una HTTP 204 (Sin contenido) junto con una carga útil vacía.

### **Servicio bidireccional**

Devuelve el contenido del mensaje de respuesta en formato JSON o, si no había ningún mensaje de respuesta disponible, un cuerpo de respuesta vacío con un código de respuesta HTTP 204 (Sin contenido).

## **Cabeceras HTTP**

Al emitir una POST HTTP, en un servicio unidireccional o bidireccional, el cuerpo HTTP debe ser un objeto JSON válido. Debe especificar:

- Una cabecera HTTP Content-Type=application/json para HTTP\_POST
- Una codificación de UTF-8

También se pueden especificar varias cabeceras HTTP opcionales, en función del verbo HTTP utilizado, para ajustar el comportamiento de MQ Service Provider. Consulte [Cabeceras HTTP que se pueden utilizar con MQ Service Provider](#) para obtener más detalles.

## **Códigos HTTP**

Si MQ Service Provider detecta un error, el servicio establece un código de estado HTTP en el rango de 400 a 599. De lo contrario, siempre se devuelven los códigos de estado documentados anteriormente.

### **Error del lado del cliente**

Si la aplicación de llamada pasa datos no válidos, se devuelve un código de estado HTTP en el rango 400-499 al interlocutor. El código de estado exacto depende del error.

Si una excepción que se está generando detecta los datos no válidos, se proporciona un rastreo de pila en formato JSON como en el ejemplo del lado del servidor.

## Error del lado del servidor

Si MQ Service Provider experimenta un error inesperado cuando el rastreo de pila se convierte al formato JSON y se devuelve al interlocutor con un código de estado HTTP de 500 (Error de servidor interno). También se grabarán los diagnósticos adecuados en los registros de z/OS Connect.

Una carga útil de respuesta de error de ejemplo simplificado es:

```
{
  "errorText": "CSQZ0006E: An unexpected JMSEException occurred while processing a request
for the 'mq7' service. ",
  "exceptionData": {
    "exceptions": [
      {
        "stackTrace": [
          "com.ibm.msg.client.jms.DetailedInvalidDestinationException: JMSWMQ2008:
Failed to open MQ queue 'ThisQueueDoesNotExist'.",
          <further content removed for brevity>
        ]
      },
      {
        "stackTrace": [
          "com.ibm.mq.MQException: JMSCMQ0001: WebSphere MQ call failed with
compcode '2' ('MQCC_FAILED') reason '2085' ('MQRC_UNKNOWN_OBJECT_NAME').",
          <further content removed for brevity>
        ]
      }
    ]
  }
}
```

## **Obtención de los MQ Service Provider**

El MQ Service Provider está soportado para su uso en IBM MQ for z/OS 8.0 y posteriores. Hay dos maneras de obtener MQ Service Provider para utilizarlo con cualquier versión soportada de IBM MQ.

1. [“Obtener la característica MQ Service Provider de Unix Systems Services Components de IBM MQ 9.1.0 \(o posterior\)” en la página 562](#)
2. [“Obtención de MQ Service Provider en Fix Central” en la página 562](#)

## **Obtener la característica MQ Service Provider de Unix Systems Services Components de IBM MQ 9.1.0 (o posterior)**

Instale la característica siguiendo las instrucciones del Directorio de programas. El Directorio de programas para IBM MQ for z/OS se puede descargar de [IBM Publications Center](#) (consulte [Archivos PDF de IBM MQ for z/OS Program Directory](#)).

MQ Service Provider estará ubicado en el directorio `PATHPREFIX/zosconnect` del sistema de archivos, donde `PATHPREFIX` es el valor elegido en el momento de la instalación. De forma predeterminada, para IBM MQ 9.1.0, `PATHPREFIX` es `/usr/lpp/mqm/V9R1M0`.

## **Obtención de MQ Service Provider en Fix Central**

Vaya a [Fix Central](#) y localice el MQ Service Provider buscando `IBM-MQ-zOSConnect-ServiceProvider`.

Esto debería localizar un archivo tar, por ejemplo, `9.1.0-IBM-MQ-zOSConnect-ServiceProvider.tar`, que se puede descargar en la estación de trabajo.



**Atención:** Se utiliza el mismo archivo tar con IBM MQ 8.0 y, también, IBM MQ 9.0. No hay ningún archivo tar separado para IBM MQ 8.0 o IBM MQ 9.0.

Una vez descargado, transfiera el archivo tar en un directorio `PATHPREFIX` en una LPAR de z/OS apropiada. Este directorio debe ser un lugar que sea accesible para z/OS Connect.

Extraiga los archivos emitiendo el siguiente mandato:

```
tar -oxvf 9.1.0-IBM-MQ-zOSConnect-ServiceProvider.tar
```

Cambie los permisos de la estructura de directorios como sea adecuado para su empresa.

## Estructura de directorios

Los dos modos de obtener el MQ Service Provider crean la misma estructura de directorios. El directorio raíz de esta estructura se conoce como *MQSP\_ROOT* en toda esta documentación, donde *MQSP\_ROOT* es *PATHPREFIX/zosconnect*.

Bajo *MQSP\_ROOT* se encuentran los archivos y directorios siguientes:

mqzosconnect.properties	Properties file that can be copied into z/OS Connect EE	
v2.0/ Connect EE	Directory containing MQ Service Provider for z/OS	
lib/ Connect EE	com.ibm.mq.zosconnect_2.0.0.jar	Feature jar file for MQ Service Provider for z/OS
features/ Connect EE	zosConnectMQ-2.0.mf	Feature manifest for MQ Service Provider for z/OS

## z/OS V9.1.0 Consideraciones sobre la transacción

HTTP no es un protocolo transaccional, por lo tanto no es posible ninguna coordinación transaccional de las operaciones de mensajería realizadas por MQ Service Provider.

Esto tiene las siguientes implicaciones:

- Si se realiza un HTTP POST para un servicio unidireccional y la conexión falla antes de que el cliente reciba una respuesta HTTP, el cliente no puede determinar inmediatamente si el mensaje se ha enviado a la cola o tema configurado o no.
- Si se realiza un HTTP DELETE para un servicio unidireccional y la conexión falla antes de que el cliente reciba una respuesta HTTP, es posible que se haya obtenido un mensaje de forma destructiva de la cola y se haya perdido, ya que no existe ninguna manera de retrotraer la obtención destructiva.
- Si se realiza un HTTP POST para un servicio bidireccional y la conexión falla antes de que el cliente reciba una respuesta HTTP, el cliente no puede indicar dónde se ha producido la anomalía. El mensaje de solicitud puede haberse enviado a la cola de solicitud o el mensaje de respuesta puede haberse obtenido de la cola de respuesta y haberse perdido.
- No hay ninguna forma de coordinar el resultado de varios verbos HTTP para un servicio unidireccional o bidireccional.

## z/OS V9.1.0 IBM z/OS Connect EE - Procedimiento de configuración de MQ Service Provider

Visión general de los componentes que hay que instalar para permitir que MQ Service Provider se configure en IBM z/OS Connect EE.

### Acerca de esta tarea

Realice las tareas siguientes, en el orden mostrado para instalar MQ Service Provider y todos los requisitos previos.

## z/OS V9.1.0 Instalación del IBM z/OS Connect EE

Los requisitos y el procedimiento para instalar IBM z/OS Connect EE.

### Antes de empezar

Asegúrese de que tiene acceso a [Update the z/OS Connect EE V2 embedded WLP](#).

## Acerca de esta tarea

Este procedimiento configura un servidor IBM z/OS Connect EE adecuado para su uso con el MQ Service Provider. Si ya tiene un servidor que se ha configurado de la misma manera, se puede utilizar ése en su lugar.

## Procedimiento

1. Cree un servidor utilizando las instrucciones descritas aquí: [Creating a z/OS Connect EE Server](#)
2. Habilite el servicio autorizado TXRRS siguiendo el procedimiento detallado en [Configuring the Liberty Angel process and z/OS authorized services](#).
3. Valide que el servicio autorizado TXRRS se ha configurado correctamente:
  - a) Iniciando el servidor.  
Siga el procedimiento detallado en [Starting and stopping z/OS Connect EE](#)
  - b) Vea los registros de servidor que están en:

```
/var/zosconnect/servers/server_name/logs/messages.log
```

Estos son archivos ASCII y deben contener una salida similar a lo siguiente:

```
A CWWKE0001I: The server test has been launched.
I CWWKB0103I: Authorized service group LOCALCOM is available.
I CWWKB0103I: Authorized service group PRODMGR is available.
I CWWKB0103I: Authorized service group SAFCRE is available.
I CWWKB0103I: Authorized service group TXRRS is available.
I CWWKB0103I: Authorized service group WOLA is available.
I CWWKB0103I: Authorized service group ZOSDUMP is available.
I CWWKB0103I: Authorized service group ZOSWLM is available.
I CWWKB0103I: Authorized service group CLIENT.WOLA is available.
I CWWKB0108I: IBM CORP product z/OS Connect version 02.00 successfully registered with z/OS
```

Compruebe la salida para ver si el grupo de servicios TXRRS autorizado está disponible. En el ejemplo anterior, la línea en negrita muestra que el grupo de servicios está disponible.

MQ Service Provider necesita que este grupo de servicios esté disponible

- c) Deteniendo el servidor.  
Siga el procedimiento detallado en [Starting and stopping z/OS Connect EE](#)

## Resultados

Ha instalado correctamente IBM z/OS Connect EE.

## Qué hacer a continuación

Instale el adaptador de recursos de IBM MQ.

### Tareas relacionadas

“IBM z/OS Connect EE - Procedimiento de configuración de MQ Service Provider” en la página 563  
Visión general de los componentes que hay que instalar para permitir que MQ Service Provider se configure en IBM z/OS Connect EE.

“Instalación del adaptador de recursos de IBM MQ” en la página 564

La característica de proveedor de mensajería IBM MQ en IBM z/OS Connect EE utiliza un componente de IBM MQ llamado adaptador de recursos IBM MQ. El adaptador de recursos se proporciona con IBM MQ como parte de las características de los componentes de IBM MQ for z/OS Unix System Services y se debe instalar antes de utilizar MQ Service Provider.

## **Instalación del adaptador de recursos de IBM MQ**

La característica de proveedor de mensajería IBM MQ en IBM z/OS Connect EE utiliza un componente de IBM MQ llamado adaptador de recursos IBM MQ. El adaptador de recursos se proporciona con IBM MQ

como parte de las características de los componentes de IBM MQ for z/OS Unix System Services y se debe instalar antes de utilizar MQ Service Provider.

## Acerca de esta tarea

**LTS** Si usted es un cliente de IBM MQ 9.1.0 Long Term Support, consulte las instrucciones de [“Configuración de la modalidad de enlaces”](#) en la página 565.

**V 9.1.1** Si los gestores de colas se van a ejecutar en la misma LPAR, o algunos de los gestores de colas se van a ejecutar en la misma LPAR, y algunos en una LPAR diferente desde el servidor IBM z/OS Connect EE, consulte las instrucciones de [“Configuración de la modalidad de enlaces”](#) en la página 565.

**V 9.1.1** Si los gestores de colas solo se van a ejecutar en LPAR diferentes desde el servidor IBM z/OS Connect EE, consulte las instrucciones de [“Configuración de la modalidad de cliente”](#) en la página 566 para utilizar conexiones de la modalidad cliente con estos gestores de colas.

## Tareas relacionadas

[“IBM z/OS Connect EE - Procedimiento de configuración de MQ Service Provider”](#) en la página 563  
Visión general de los componentes que hay que instalar para permitir que MQ Service Provider se configure en IBM z/OS Connect EE.

[“Instalación del IBM z/OS Connect EE”](#) en la página 563

Los requisitos y el procedimiento para instalar IBM z/OS Connect EE.

[“Instalación del MQ Service Provider en IBM z/OS Connect EE”](#) en la página 568

Debe instalar el MQ Service Provider en IBM z/OS Connect EE antes de que se utilice.

**z/OS** *Configuración de la modalidad de enlaces*

Instale el adaptador de recursos de IBM MQ para la conectividad de la modalidad de enlaces con IBM MQ.

## Antes de empezar

Asegúrese de que la característica de IBM MQ for z/OS Unix System Services Components, para la versión del gestor de cola al que se va a conectar, está instalada.

Si va a conectarse a varias versiones del gestor de colas, debe utilizar la característica de la última versión.

## Acerca de esta tarea

Este procedimiento instala el adaptador de recursos IBM MQ en IBM z/OS Connect EE.

## Procedimiento

1. Localice el directorio de Unix System Services Components de la versión de IBM MQ a la que se va a conectar.

Por ejemplo, este directorio podría llamarse `/usr/lpp/mqm/V9R1M0/`. El directorio contiene un subdirectorio `java/lib`, que contiene un número de bibliotecas nativas (archivos `.so`).



**Atención:** Si va a conectarse a varias versiones del gestor de colas, utilice la última versión.

2. Edite el archivo `server.xml` del servidor IBM z/OS Connect EE.

Añada las líneas siguientes:

```
<variable name="wmqJmsClient.rar.location"
  value="MQJAVA_LIB_DIR/jca/wmq.jmsra.rar"/>
<wmqJmsClient nativeLibraryPath="MQJAVA_LIB_DIR"/>
```

donde `MQJAVA_LIB_DIR` se basa en el directorio situado en el paso “1” en la página 565, por ejemplo, `/usr/lpp/mqm/V9R1M0/java/lib`.

b. Guarde los cambios.

La primera línea indica a IBM z/OS Connect EE dónde localizar el adaptador de recursos IBM MQ.

La segunda línea indica al adaptador de recursos de IBM MQ dónde debe localizar las bibliotecas nativas que se utilizan para las conexiones de enlaces a IBM MQ.

Consulte [wmqJmsClient](#) y [Deploying JMS applications to Liberty to use the IBM MQ messaging provider](#) para obtener más información.

3. Configure la STEPLIB del servidor para incluir bibliotecas de IBM MQ.

Debe hacerlo para que las bibliotecas nativas ubicadas en el paso “1” en la página 565 puedan conectarse al gestor de colas.

Normalmente esto implica la edición del JCL, que inicia el servidor, para incluir lo siguiente:

```
//STEPLIB DD DSN=HLQ.SCSQAUTH,DISP=SHR
// DD DSN=HLQ.SCSQANLE,DISP=SHR
```

donde `HLQ` es el calificador de alto nivel de los conjuntos de datos que contienen la instalación de IBM MQ.

Si va a conectarse a varias versiones de IBM MQ desde el mismo servidor, utilice los conjuntos de datos de la versión más reciente.

## Resultados

Ha instalado parcialmente el adaptador de recursos de IBM MQ.

Tenga en cuenta que el adaptador de recursos IBM MQ no se instalará por completo hasta que su característica padre (`wmqJmsClient-2.0`) se habilite, tal como se describe en [“Habilitación de IBM z/OS Connect EE y el MQ Service Provider”](#) en la página 568.

## Qué hacer a continuación

Instale MQ Service Provider en IBM z/OS Connect EE.

### Tareas relacionadas


[“IBM z/OS Connect EE - Procedimiento de configuración de MQ Service Provider”](#) en la página 563  
Visión general de los componentes que hay que instalar para permitir que MQ Service Provider se configure en IBM z/OS Connect EE.

[“Instalación del IBM z/OS Connect EE”](#) en la página 563

Los requisitos y el procedimiento para instalar IBM z/OS Connect EE.

[“Instalación del MQ Service Provider en IBM z/OS Connect EE”](#) en la página 568

Debe instalar el MQ Service Provider en IBM z/OS Connect EE antes de que se utilice.

 Configuración de la modalidad de cliente

Instale el adaptador de recursos IBM MQ para la conectividad de la modalidad de cliente con IBM MQ.

## Antes de empezar

Asegúrese de que está instalada la característica de los componentes de IBM MQ for z/OS Unix System Services (USS), para la versión del gestor de colas al que se está intentando conectar.

De forma alternativa, obtenga el adaptador de recursos IBM MQ desde IBM [Fix Central](#), buscando *Adaptador de recursos MQ*.

Utilice la versión del adaptador de recursos que coincida con la versión de IBM MQ al que se está conectando. Copie el archivo en el sistema de archivos USS de z/OS que contiene la instalación del

servidor IBM z/OS Connect EE y extráigalo en un directorio de su elección, siguiendo las instrucciones proporcionadas con la descarga.

Si se va a conectar a varias versiones del gestor de colas, debería utilizar la característica o el adaptador de recursos de la última versión.

## Acerca de esta tarea

Este procedimiento instalar el adaptador de recursos IBM MQ en IBM z/OS Connect EE.

## Procedimiento

1. Localice el directorio de Unix System Services Components de la versión de IBM MQ a la que se va a conectar.

Por ejemplo, este directorio podría llamarse `/usr/lpp/mqm/V9R1M0/`.

Localice de forma alternativa el directorio en el que ha descargado el adaptador de recursos IBM MQ. Por ejemplo, `/usr/lpp/mqm/ra/V9R1M0/`.

2. Edite el archivo `server.xml` del servidor IBM z/OS Connect EE.

Añada las líneas siguientes:

```
a. <variable name="wmqJmsClient.rar.location"
           value="MQRA_DIR/wmq.jmsra.rar"/>
```

donde `MQRA_DIR` se basa en el directorio ubicado en el paso [“1” en la página 567](#)

- b. Guarde los cambios.

La variable indica a IBM z/OS Connect EE dónde localizar el adaptador de recursos IBM MQ.

Consulte [wmqJmsClient](#) y [Deploying JMS applications to Liberty to use the IBM MQ messaging provider](#) para obtener más información.

## Resultados

Ha instalado parcialmente el adaptador de recursos de IBM MQ.

Tenga en cuenta que el adaptador de recursos IBM MQ no se instalará por completo hasta que su característica padre (`wmqJmsClient-2.0`) se habilite, tal como se describe en [“Habilitación de IBM z/OS Connect EE y el MQ Service Provider” en la página 568](#).

## Qué hacer a continuación

Instale MQ Service Provider en IBM z/OS Connect EE.

### Tareas relacionadas

[“IBM z/OS Connect EE - Procedimiento de configuración de MQ Service Provider” en la página 563](#)

Visión general de los componentes que hay que instalar para permitir que MQ Service Provider se configure en IBM z/OS Connect EE.

[“Instalación del IBM z/OS Connect EE” en la página 563](#)

Los requisitos y el procedimiento para instalar IBM z/OS Connect EE.

[“Instalación del adaptador de recursos de IBM MQ” en la página 564](#)

La característica de proveedor de mensajería IBM MQ en IBM z/OS Connect EE utiliza un componente de IBM MQ llamado adaptador de recursos IBM MQ. El adaptador de recursos se proporciona con IBM MQ como parte de las características de los componentes de IBM MQ for z/OS Unix System Services y se debe instalar antes de utilizar MQ Service Provider.

[“Instalación del MQ Service Provider en IBM z/OS Connect EE” en la página 568](#)

Debe instalar el MQ Service Provider en IBM z/OS Connect EE antes de que se utilice.



Debe instalar el MQ Service Provider en IBM z/OS Connect EE antes de que se utilice.

## Acerca de esta tarea

Para ello debe realizar el procedimiento siguiente:

## Procedimiento

1. Localice el directorio de las extensiones del producto IBM z/OS Connect EE.  
Para una instalación típica, este es el directorio `/var/zosconnect/v2r0/extensions` que se describe aquí: [Configuración del directorio de extensiones del producto](#).
2. Copie `MQSP_ROOT/mqzosconnect.properties` en el directorio ubicado en el paso “1” en la [página 568](#).  
Consulte “Estructura de directorios” en la [página 563](#) si desea más información sobre `MQSP_ROOT`.
3. Edite el archivo copiado.  
Tenga en cuenta que este archivo es un archivo ASCII. Cambie `PATH_TO_INSTALL` a `MQSP_ROOT/v2.0` y guarde los cambios.



**Atención:** El paso “3” en la [página 568](#) presupone que el ID de usuario que ejecuta el servidor tiene acceso de lectura a la estructura de directorios de `MQSP_ROOT`. Si éste no es el caso, añada suficiente acceso al ID de usuario o mueva el contenido de `MQSP_ROOT` a una ubicación con suficiente acceso.

## Resultados

Ahora ya ha instalado todos los requisitos previos para MQ Service Provider.

## Qué hacer a continuación

Ahora debe habilitar el MQ Service Provider y IBM z/OS Connect EE.

### Tareas relacionadas

“IBM z/OS Connect EE - Procedimiento de configuración de MQ Service Provider” en la [página 563](#)  
Visión general de los componentes que hay que instalar para permitir que MQ Service Provider se configure en IBM z/OS Connect EE.

“Instalación del adaptador de recursos de IBM MQ” en la [página 564](#)

La característica de proveedor de mensajería IBM MQ en IBM z/OS Connect EE utiliza un componente de IBM MQ llamado adaptador de recursos IBM MQ. El adaptador de recursos se proporciona con IBM MQ como parte de las características de los componentes de IBM MQ for z/OS Unix System Services y se debe instalar antes de utilizar MQ Service Provider.

“Habilitación de IBM z/OS Connect EE y el MQ Service Provider” en la [página 568](#)

Qué debe hacer para habilitar IBM z/OS Connect EE y el MQ Service Provider.

Qué debe hacer para habilitar IBM z/OS Connect EE y el MQ Service Provider.

## Antes de empezar

Asegúrese de que ha completado los procedimientos:

- “[Instalación del IBM z/OS Connect EE](#)” en la [página 563](#)
- “[Instalación del adaptador de recursos de IBM MQ](#)” en la [página 564](#)
- “[Instalación del MQ Service Provider en IBM z/OS Connect EE](#)” en la [página 568](#)



## Acerca de esta tarea

Este procedimiento habilita tanto IBM z/OS Connect EE como el MQ Service Provider.

## Procedimiento

1. Edite el archivo `server.xml` del servidor IBM z/OS Connect EE que ha creado y sustituya todo el elemento **featureManager** con estas líneas:

```
<featureManager>
  <feature>zosconnect:zosconnect-2.0</feature>
  <feature>appSecurity-2.0</feature>
  <feature>jms-2.0</feature>
  <feature>mqzosconnect:zosConnectMQ-2.0</feature>
  <feature>wmqJmsClient-2.0</feature>
  <feature>zosTransaction-1.0</feature>
</featureManager>
```



**Atención:** Corrija estas entradas, sólo si aún no existen.

2. Asegúrese de que el ID de usuario con el que se ejecuta el servidor IBM z/OS Connect EE tenga acceso ALTER a MVSADMIN.RRS.COMMANDS.\* en la clase RACF FACILITY.
3. Configure la seguridad para IBM z/OS Connect EE.  
Consulte [Protección de los recursos de z/OS Connect EE](#) para obtener detalles de cómo realizar este procedimiento.
4. Inicie el servidor.

## Qué hacer a continuación

Compruebe que IBM z/OS Connect EE se ha configurado correctamente.

### Tareas relacionadas

“IBM z/OS Connect EE - Procedimiento de configuración de MQ Service Provider” en la página 563  
Visión general de los componentes que hay que instalar para permitir que MQ Service Provider se configure en IBM z/OS Connect EE.

“Comprobación de que IBM z/OS Connect EE está configurado correctamente” en la página 569  
Cómo se comprueba que IBM z/OS Connect EE se ha configurado correctamente.

## **Comprobación de que IBM z/OS Connect EE está configurado correctamente**

Cómo se comprueba que IBM z/OS Connect EE se ha configurado correctamente.

## Antes de empezar

Asegúrese de que ha llevado a cabo el procedimiento detallado en [“Habilitación de IBM z/OS Connect EE y el MQ Service Provider”](#) en la página 568

## Acerca de esta tarea

IBM z/OS Connect EE proporciona una API RESTful que se puede utilizar para consultar los servicios que están instalados y realizar acciones administrativas como, por ejemplo, detenerlas e iniciarlas.

## Procedimiento

1. Emita un mandato HTTP GET en IBM z/OS Connect EE para consultar la lista de servicios instalados actualmente.  
Puede hacerlo con un navegador web especificando un URL con el formato

```
https://HOST_NAME:HTTPS_PORT/zosConnect/services
```

donde *NOMBRE\_HOST* y *PUERTO\_HTTPS* son los valores que ha especificado en el paso “1” en la página 564 de [“Instalación del IBM z/OS Connect EE”](#) en la página 563.

Por ejemplo,

```
https://yourdomainname:12342/zosConnect/services
```

2. Especifique el ID de usuario y la contraseña cuando se lo solicite el navegador.

Estos son los valores que ha especificado en el elemento de usuario en el paso “3” en la página 569 de [“Habilitación de IBM z/OS Connect EE y el MQ Service Provider”](#) en la página 568.

## Resultados

Esto produce la respuesta JSON siguiente, lo que indica que z/OS Connect está en ejecución, pero no ha obtenido los servicios instalados. Tenga en cuenta que, si se ha utilizado un servidor IBM z/OS Connect EE existente, que tenía servicios instalados, estos servicios se mostrarán.

```
-----  
{  
  "zosConnectServices": [  ]  
}
```

## Qué hacer a continuación

[Configurar un servicio de IBM MQ unidireccional simple en z/OS Connect EE.](#)

### Tareas relacionadas

[“IBM z/OS Connect EE - Procedimiento de configuración de MQ Service Provider”](#) en la página 563  
Visión general de los componentes que hay que instalar para permitir que MQ Service Provider se configure en IBM z/OS Connect EE.

## **Configuración de un servicio MQ Service Provider unidireccional simple en IBM z/OS Connect EE**

Utilice los pasos siguientes, en orden, para configurar un servicio MQ Service Provider unidireccional simple que se conecta a IBM MQ en la modalidad de enlaces.

## Antes de empezar

Asegúrese de que ha configurado todos los componentes correctamente, tal como se describe en [“Habilitación de IBM z/OS Connect EE y el MQ Service Provider”](#) en la página 568 y [“Comprobación de que IBM z/OS Connect EE está configurado correctamente”](#) en la página 569.

## Procedimiento

1. Cree una cola denominada ONE\_WAY\_QUEUE en el gestor de colas de z/OS de destino, utilizando MQSC o IBM MQ Explorer.
2. Defina una fábrica de conexiones de proveedor de mensajería de IBM MQ y una cola.  
Puede hacerlo añadiendo lo siguiente a la parte inferior de `server.xml`, del servidor IBM z/OS Connect EE que ha creado, pero dentro del elemento del servidor.

```
-----  
<jmsConnectionFactory id="cf1" jndiName="jms/cf1" connectionManagerRef="ConMgr1">  
  <properties.wmqJms  
    transportType="BINDINGS"  
    queueManager="MQ21"/>  
</jmsConnectionFactory>
```

```

</jmsConnectionFactory>
<connectionManager id="ConMgr1" maxPoolSize="5"/>
<jmsQueue id="q1" jndiName="jms/d1">
  <properties.wmqJms
    baseQueueName="ONE_WAY_QUEUE"/>
</jmsQueue>
-----

```

### Notas:

- a. Cambie el valor del atributo **queueManager** por el nombre de gestor de colas de destino correcto.
- b. **V9.1.1** En este ejemplo, se utiliza un **transportType** de BINDINGS. Esto permite que se realicen conexiones de memoria cruzada con gestores de colas que se ejecutan en la misma LPAR que el servidor IBM z/OS Connect EE.

Al utilizar el MQ Service Provider proporcionado con IBM MQ 9.1.1 o posterior, también es posible utilizar un **transportType** de CLIENT, que permite conexiones basadas en TCP/IP a gestores de colas z/OS, que se ejecutan en la misma LPAR que el servidor IBM z/OS Connect EE. o en una LPAR remota diferente.

Cuando se utiliza un **transportType** de CLIENT, se tiene que especificar un número de otras propiedades. Consulte [“Modalidades de conexión” en la página 558](#) si desea más información sobre estas propiedades.

3. Defina un servicio IBM MQ simple unidireccional añadiendo lo siguiente al `server.xml` del servidor z/OS Connect EE creado, pero dentro del elemento `server`.

```

-----
<zosconnect_zosConnectService id="zosconnMQ1"
  invokeURI="/oneWay"
  serviceName="oneWay"
  serviceRef="oneWay " />

<mqzosconnect_mqzOSConnectService id="oneWay "
  connectionFactory="jms/cf1"
  destination="jms/d1"/>
-----

```

El elemento `zosConnectService` define un nuevo servicio en IBM z/OS Connect EE utilizando el **serviceName** de `oneWay`. El:

- atributo **invokeURI** facilita la invocación del servicio.
- atributo **serviceRef** tiene que coincidir con el atributo ID de un proveedor de servicio de z/OS Connect, que en este caso lo proporciona el elemento `mqzOSConnectService`.

El elemento `mqzOSConnectService` define una única instancia de servicio proporcionada por MQ Service Provider.

Los atributos **connectionFactory** y **destination** indican a la instancia cómo localizar la fábrica de conexiones y la cola, respectivamente, del proveedor de mensajería de IBM MQ.

Consulte el elemento `mqzOSConnectService` para obtener detalles de los atributos de esta estructura.

## Resultados

Ha configurado un servicio unidireccional simple.


## Qué hacer a continuación

Necesita probar el servicio.

### Tareas relacionadas

[“Probar el servicio unidireccional en IBM z/OS Connect EE” en la página 572](#)

Serie de pasos que garantiza que el servicio unidireccional funciona.

 Probar el servicio unidireccional en IBM z/OS Connect EE  
Serie de pasos que garantiza que el servicio unidireccional funciona.

## Antes de empezar

Asegúrese de que ha completado “[Configuración de un servicio MQ Service Provider unidireccional simple en IBM z/OS Connect EE](#)” en la [página 570](#) correctamente.

## Procedimiento

1. Compruebe que z/OS Connect reconoce el nuevo servicio.

Hágalo volviendo a emitir los pasos detallados en “[Comprobación de que IBM z/OS Connect EE está configurado correctamente](#)” en la [página 569](#).

Dado que ahora hay un servicio definido, debe ver algo similar a la salida siguiente:

```
-----  
{  
  "zosConnectServices": [  
    {  
      "ServiceName": "oneWay",  
      "ServiceDescription": "DATA_UNAVAILABLE",  
      "ServiceProvider": "IBM MQ for z/OS service provider for IBM z/OS Connect EE V2.0",  
      "ServiceURL": "https://yourdomainname:12342/zosConnect/services/oneWay"  
    }  
  ]  
}
```

**Nota: ServiceDescription** se devuelve como DATA\_UNAVAILABLE porque el atributo **serviceDescription** no se ha especificado en el elemento zosConnectService.

2. Emita un HTTP POST para el servicio. Esto hace que se envíe un nuevo mensaje a ONE\_WAY\_QUEUE con la carga útil que se haya especificado en el POST.

Una manera correcta de realizar este tipo de prueba es utilizando algo parecido al plug-in de [Cliente REST avanzado \(ARC\)](#) en Google Chrome.

- a) Utilice ARC para emitir un POST HTTP en el URL siguiente:

```
https://HOST_NAME:HTTPS_PORT/oneWay
```

- b) Especifique un JSON válido para el cuerpo, por ejemplo:

```
{"name1" : "value1", "name2" : "value2"}
```

Debe especificar dos cabeceras:

- i) Autorización = Basic *ENCODED\_USERID\_PASSWORD*

donde *ENCODED\_USERID\_PASSWORD* es un ID de usuario y una contraseña codificados en base 64 tal como se describe en [Esquema de autenticación básica](#).

- ii) Content-Type = application/json

La primera cabecera se utiliza para iniciar la sesión en z/OS Connect y es una forma codificada del ID de usuario y la contraseña utilizados en “[Habilitación de IBM z/OS Connect EE y el MQ Service Provider](#)” en la [página 568](#).

La segunda cabecera informa al servidor de que la carga útil es JSON, que es el único valor soportado al emitir llamadas HTTP POST a MQ Service Provider.

Si no especifica esta cabecera, se produce un error HTTP 415, que indica un tipo no soportado.

Al emitir un POST HTTP para un servicio unidireccional hace que se devuelvan un cuerpo de respuesta HTTP vacío y un código de respuesta HTTP de 204 (sin contenido).

ARC genera un mensaje indicando que la respuesta no contiene datos.

Para validar que se ha puesto un mensaje, utilice una herramienta como IBM MQ Explorer para examinar el contenido de ONE\_WAY\_QUEUE. Debe haber un mensaje IBM MQ de formato MQSTR en la cola con la carga útil JSON que se ha enviado.

3. Utilice ARC para emitir un HTTP DELETE para el URL siguiente:

```
https://HOST_NAME:HTTPS_PORT/oneWay
```

Si se hace esto se obtiene de forma destructiva el mensaje que se acaba de enviar al servicio unidireccional.

Esto debe dar un cuerpo de respuesta que contenga la carga útil SON que se ha enviado originalmente. Puede utilizar IBM MQ Explorer para confirmar que el mensaje se ha obtenido de forma destructiva.

## **Configuración de un servicio IBM MQ bidireccional simple en IBM z/OS Connect EE**

Cómo se configura un servicio IBM MQ bidireccional en IBM z/OS Connect EE, con respecto a un par existente de colas IBM MQ llamadas MQ\_REQUEST\_Q y MQ\_REPLY\_Q.

### Antes de empezar

Asegúrese de que ha configurado todos los componentes correctamente, tal como se describe en [“Habilitación de IBM z/OS Connect EE y el MQ Service Provider”](#) en la página 568 y [“Comprobación de que IBM z/OS Connect EE está configurado correctamente”](#) en la página 569.

### Acerca de esta tarea

Esto implica algo más que configurar un servicio unidireccional, ya que requiere que una aplicación de fondo consuma los mensajes de solicitud enviados utilizando z/OS Connect, y generar mensajes de respuesta.

Esta tarea presupone que las colas MQ\_REQUEST\_Q y MQ\_REPLY\_Q son utilizadas por una aplicación de fondo existente, por ejemplo, una transacción CICS o IMS, que toma un mensaje de solicitud de MQ\_REQUEST\_Q y genera un mensaje de respuesta que se coloca en MQ\_REPLY\_Q.

### Procedimiento

1. Defina una fábrica de conexiones del proveedor de mensajería IBM MQ y dos colas de proveedor de mensajería IBM MQ en el servidor IBM z/OS Connect EE.

Las colas del proveedor de mensajería IBM MQ se correlacionan con las colas MQ\_REQUEST\_Q y MQ\_REPLY\_Q existentes. Puede hacerlo añadiendo lo siguiente a la parte inferior de `server.xml`, del servidor IBM z/OS Connect EE que ha creado, pero dentro del elemento del servidor.

```
<jmsConnectionFactory id="cf2" jndiName="jms/cf2"
  connectionManagerRef="ConMgr2">
  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21"/>
</jmsConnectionFactory>

<connectionManager id="ConMgr2" maxPoolSize="5"/>

<jmsQueue id="requestQueue" jndiName="jms/requestQueue">
  <properties.wmqJms baseQueueName="MQ_REQUEST_Q"/>
</jmsQueue>

<jmsQueue id="replyQueue" jndiName="jms/replyQueue">
```

```
<properties.wmqJms baseQueueName="MQ_REPLY_Q"/>
</jmsQueue>
```

### Notas:

- a. Cambie el valor del atributo **queueManager** por el nombre de gestor de colas de destino correcto.
- b. Es posible que tenga que realizar la configuración adicional de estos elementos, en función de la naturaleza de la aplicación de fondo. Consulte [“Configuración de MQ Service Provider”](#) en la página 576 para obtener más información.
- c. El atributo **maxPoolSize** podría tener que ajustarse en función del número de solicitudes simultáneas. Si desea más información sobre este atributo, consulte [Configuración de la técnica de agrupación de conexiones para conexiones JMS](#).
- d. **V9.1.1** En este ejemplo, se utiliza un **transportType** de BINDINGS. Esto permite que se realicen conexiones de memoria cruzada con gestores de colas que se ejecutan en la misma LPAR que el servidor IBM z/OS Connect EE.

Al utilizar el MQ Service Provider proporcionado con IBM MQ 9.1.1 o posterior, también es posible utilizar un **transportType** de CLIENT, que permite conexiones basadas en TCP/IP a gestores de colas z/OS, que se ejecutan en la misma LPAR que el servidor IBM z/OS Connect EE. o en una LPAR remota diferente.

Cuando se utiliza un **transportType** de CLIENT, se tiene que especificar un número de otras propiedades. Consulte [“Modalidades de conexión”](#) en la página 558 si desea más información sobre estas propiedades.

2. Defina el servicio IBM MQ bidireccional añadiendo lo siguiente al `server.xml` del servidor z/OS Connect EE creado, pero dentro del elemento `server`.

```
-----
<zosconnect_zosConnectService id="zosconnMQ2"
    invokeURI="/twoWay"
    serviceName="twoWay"
    serviceRef="twoWay" />

<mqzosconnect_mqzOSConnectService id="twoWay"
    connectionFactory="jms/cf2"
    destination="jms/requestQueue"
    replyDestination="jms/replyQueue"
    waitInterval="10000"/>
-----
```

### Notas:

- a. Es posible que tenga que realizar la configuración adicional de estos elementos, en función de la naturaleza de la aplicación de fondo. Por ejemplo, podría necesitar transformación de datos. Consulte [“Configuración de MQ Service Provider”](#) en la página 576 para obtener más información.
- b. Es posible que tenga que ajustar el valor de **waitInterval**, consulte [waitInterval](#) para obtener más información.
- c. En este ejemplo, se da por supuesto que los mensajes colocados en el destino de respuesta por la aplicación de fondo tienen un ID de correlación que es el mismo que el ID del mensaje en el destino de solicitud.

Si este no es el caso, tendrá que añadir el atributo **replySelection** al elemento `mqzosconnect_mqzOSConnectService` y establezca el valor de forma apropiada.

Consulte [replySelection](#) para obtener más información.

## Resultados

Ha configurado un servicio bidireccional en z/OS Connect EE, que se puede utilizar con una aplicación de fondo existente que utiliza un par de consultas IBM MQ.

## Qué hacer a continuación

Puede invocar el servicio bidireccional emitiendo un mandato HTTP POST al URL `https://host_name:port_no/twoWay`.

Los pasos para realizar esto son similares a los descritos en [“Probar el servicio unidireccional en IBM z/OS Connect EE”](#) en la página 572, excepto que solo está soportado el verbo HTTP POST y deberá enviar datos de carga útil JSON que sean apropiados para la aplicación de fondo existente.

### Tareas relacionadas

[“Configuración de un servicio MQ Service Provider unidireccional simple en IBM z/OS Connect EE”](#) en la página 570

Utilice los pasos siguientes, en orden, para configurar un servicio MQ Service Provider unidireccional simple que se conecta a IBM MQ en la modalidad de enlaces.

## Generación de un archivo de archivado de servicio (SAR)

Antes de poder crear una API con IBM z/OS Connect EE, tendrá que generar un archivo SAR.

### Acerca de esta tarea

Puede conseguirlo con IBM MQ, utilizando el proceso CICS-WOLA descrito en la información de *IBM z/OS Connect EE* en IBM Documentation. Consulte [Generar un archivo de servicio a partir del libro de copias CICS COBOL](#) si desea más información.

**Importante:** El proceso descrito en el texto siguiente es opcional. Si no desea utilizar los recursos proporcionados por el editor de API, no necesita el archivo SAR. Sin embargo, es posible que desee utilizar los esquemas JSON y los archivos de enlace que se describen en el enlace anterior, si desea utilizar el soporte de transformación de datos proporcionado por IBM z/OS Connect EE.

El proceso descrito en el enlace anterior utiliza el programa de utilidad **BAQLS2JS** proporcionado con IBM z/OS Connect EE.

Este programa de utilidad:

1. Genera esquemas JSON y archivos de enlace que se pueden utilizar con la función de transformación de datos proporcionada en z/OS Connect EE.

Si desea configurar estos archivos en IBM z/OS Connect EE, debe establecerlos en la configuración a través del elemento `zosconnect_zosConnectDataXform`.

Consulte [“Utilización de transformaciones de datos con MQ Service Provider”](#) en la página 578 para obtener más información.

2. Genera un archivo SAR que contiene los esquemas JSON del paso anterior y una referencia al atributo **serviceName** de una instancia de servicio IBM z/OS Connect EE particular.

La referencia se proporciona en **BAQLS2JS** utilizando el parámetro `SERVICE-NAME`.

Por ejemplo, si `SERVICE-NAME=MQ1WayBackend` está establecido como un atributo en **BAQLS2JS**, **BAQLS2JS** generará un SAR que se podría utilizar con una instancia de servicio de IBM MQ configurada de la forma siguiente:

```
<zosconnect_zosConnectService id="MQ1WayService"
    serviceName="MQ1WayBackend"
    serviceRef="mq1way"
    invokeURI="/mq1way"
    dataXformRef="xfoimJSON2Byte" />

<mqzosconnect_mqzOSConnectService id="mq1way"
    connectionFactory="jms/cf1"
    destination="jms/oneWayRequestQ" />
```

## Resultados

Una vez generado, el archivo SAR se puede importar en el editor de API para generar una API. Consulte [Crear una API utilizando Editor de API de z/OS Connect EE](#) si desea más información.

Una vez que se ha construido una API en el editor de API, se genera un archivo de API (AAR) que, después, se puede desplegar en z/OS Connect EE, donde se ejecutará en el servicio referenciado por el SERVICE-NAME atribuido en **BAQLS2JS**. Consulte [Despliegue de una API en el servidor z/OS Connect EE](#) si desea más información.

### Tareas relacionadas

[“Configuración de un servicio MQ Service Provider unidireccional simple en IBM z/OS Connect EE” en la página 570](#)

Utilice los pasos siguientes, en orden, para configurar un servicio MQ Service Provider unidireccional simple que se conecta a IBM MQ en la modalidad de enlaces.

[“Configuración de un servicio IBM MQ bidireccional simple en IBM z/OS Connect EE” en la página 573](#)  
Cómo se configura un servicio IBM MQ bidireccional en IBM z/OS Connect EE, con respecto a un par existente de colas IBM MQ llamadas MQ\_REQUEST\_Q y MQ\_REPLY\_Q.

## Configuración de MQ Service Provider

Existen varias áreas que debe tener en cuenta, antes de utilizar MQ Service Provider.

Esta sección incluye:

- [“Consideraciones sobre seguridad al utilizar MQ Service Provider” en la página 576](#)
- [“Utilización de transformaciones de datos con MQ Service Provider” en la página 578](#)
- [“Trabajar con aplicaciones MQI” en la página 582](#)

## Consideraciones sobre seguridad al utilizar MQ Service Provider

Hay dos conjuntos de consideraciones sobre seguridad para MQ Service Provider.

- Si un usuario determinado puede enviar solicitudes RESTful a los servicios MQ Service Provider expuestos por IBM z/OS Connect EE.
- Si un servicio de MQ Service Provider determinado está autorizado a acceder al gestor de colas y sus recursos.

### Si un usuario determinado puede enviar solicitudes RESTful a servicios MQ Service Provider expuestos de IBM z/OS Connect EE

Esto se controla mediante la configuración en el archivo `server.xml` de IBM z/OS Connect EE.

Consulte [Configuración de la seguridad para z/OS Connect EE](#).

### Si un servicio de MQ Service Provider determinado está autorizado a acceder al gestor de colas y sus recursos

MQ Service Provider es fundamentalmente una aplicación JMS, basada en el proveedor de mensajería de IBM MQ que viene con WLP, que se conecta a uno o varios gestores de colas utilizando conexiones de modalidad de enlaces.

Como resultado, MQ Service Provider puede asegurarse de la misma manera que las aplicaciones que comparten estas características. Este tema describe estas similitudes y señala las diferencias.

Puesto que MQ Service Provider es una aplicación JMS, se conecta a un gestor de colas y presenta un ID de usuario y una contraseña opcional. Opcionalmente, éstos se validan utilizando la [Autenticación de conexión](#).

Si es válido, se permite al usuario conectarse dependiendo de la configuración de seguridad del gestor de colas. Consulte [Configuración de la seguridad en z/OS para obtener más información](#).



Los atributos de configuración del elemento de servicio `mzOSConnect` y el subelemento `properties.wmqJMS` del elemento `mqzOSConnectFactory` en `server.xml` afectan a qué ID de usuario y contraseña opcional se presentan al gestor de colas.

Las diversas posibilidades se detallan en la tabla siguiente.



**Atención:** Existe un orden de prioridad entre las propiedades de configuración de seguridad. Los atributos `mqzOSConnectService` alteran temporalmente los atributos `properties.wmqJMS` y el atributo `useCallerPrincipal` alteran temporalmente todos los demás atributos.

Ambos atributos de contraseña se pueden proporcionar en texto sin formato o formato codificado. Debe utilizar el formato codificado, ya que cualquiera con acceso a `server.xml` puede ver la contraseña en texto sin formato.

IBM z/OS Connect EE se envía con una herramienta llamada `securityUtility`, que se puede utilizar para codificar contraseñas. Consulte [Liberty: securityUtility command](#) para obtener más información.

*Tabla 35. Autorizaciones de servicio*

Elemento <code>mqzOSConnectService</code>		Elemento <code>properties.wmqJMS</code>		Resultado
<code>useCallerPrincipal</code>	<code>userName and password</code>	<code>userName</code>	<code>password</code>	
no establecido/ falso	no establecido/en blanco	no establec ido/en blanco	no establec ido/en blanco	El nombre de usuario asociado al espacio de direcciones de IBM z/OS Connect EE se presenta al gestor de colas con fines de autorización y autenticación. No se presenta ninguna contraseña.
no establecido/ falso	no establecido/en blanco	set	no establec ido/en blanco	El nombre de usuario en el elemento <code>properties.wmqJMS</code> se presenta al gestor de colas para autenticación y autorización. No se presenta ninguna contraseña.
no establecido/ falso	no establecido/en blanco	set	set	El nombre de usuario y la contraseña en el elemento <code>properties.wmqJMS</code> se presentan al gestor de colas para autenticación y autorización.
no establecido/ falso	ambos valores establecidos	se ignora si está establec ido	se ignora si está establec ido	El nombre de usuario y la contraseña en el elemento <code>mqzOSConnectService</code> se presentan al gestor de colas para autenticación y autorización.
true	se ignora si está establecido	se ignora si está establec ido	se ignora si está establec ido	Se toma el principal de usuario que se ha autenticado en IBM z/OS Connect EE, y su nombre de usuario se presenta al gestor de colas con fines de autorización y autenticación. No se presenta ninguna contraseña.  Aquí se presenta la información sobre cómo configurar la seguridad para IBM z/OS Connect EE. <sup>1</sup>

**Nota:**

## 1. Configuring security for z/OS Connect EE

### Ejemplos

Esto muestra un servicio unidireccional donde la fábrica de conexiones y `mqzOSConnectService` especifican un **userID** y **password**. En ambos casos las contraseñas están codificadas. Se utilizan el **userID** y **password** que se definen en la definición `mqzOSConnectService`.

#### IBM z/OS Connect EE

```
<zoscconnect_zosConnectService
id="samplezOSConnectService1"
invokeURI="/samplezOSConnectService1"
serviceName="samplezOSConnectService1_name"
serviceRef="samplezOSConnectService1_MQ" />

<mqzoscconnect_mqzOSConnectService
id="samplezOSConnectService1_MQ"
connectionFactory="jms/sampleCF1"
userName="bill"
password="{aes}AJ+DdZ+1u0KEG5KIwUz4LvHBAQ8nTd3y8K8HAI+48Tt"
destination="jms/sampleQ1" />

<jmsConnectionFactory
id="sampleCF1"
jndiName="jms/sampleCF1"
connectionManagerRef="sampleCF2ConnectionManager1">

<properties.wmqJms
transportType="BINDINGS"
queueManager="MQ21"
userName="matt"
password="{xor}GBMeEg9uERg=" />
</jmsConnectionFactory>

<jmsQueue
id="sampleQ1"
jndiName="jms/sampleQ1">

<properties.wmqJms
baseQueueName="SampleQ1" />
</jmsQueue>
```

Este ejemplo muestra una definición de servicio unidireccional que utiliza los mismos objetos JMS (la cola `sampleQ1` y la fábrica de conexiones `sampleCF1`) como el primer ejemplo. El principal que se autentica en IBM z/OS Connect EE se presenta al gestor de colas porque se ha especificado **useCallerPrincipal="true"**.

#### IBM z/OS Connect EE

```
<zoscconnect_zosConnectService
id="samplezOSConnectService2"
invokeURI="/samplezOSConnectService2"
serviceName="samplezOSConnectService2_name"
serviceRef="samplezOSConnectService2_MQ" />

<mqzoscconnect_mqzOSConnectService
id="samplezOSConnectService2_MQ"
connectionFactory="jms/sampleCF1"
destination="jms/sampleQ1"
useCallerPrincipal="true" />
```

### Utilización de transformaciones de datos con MQ Service

#### Provider

IBM z/OS Connect EE proporciona la capacidad de transformar los datos JSON a un formato arbitrario antes de llamar a un activo z/OS de fondo como, por ejemplo, una transacción CICS, y transformar la respuesta del activo z/OS de nuevo a JSON.

Esta capacidad la proporcionan los proveedores de transformación de datos conectables. Un proveedor incorporado permite que JSON se transforme a y desde estructuras COBOL, PLI o C.

MQ Service Provider funciona con transformaciones de datos de IBM z/OS Connect EE, pero existe una serie de consideraciones que debería tener en cuenta.

## Más información sobre transformaciones de datos IBM z/OS Connect EE

Consulte [Definición de transformadores de datos](#).

### Envío de mensajes a IBM MQ

Cuando un servicio de MQ Service Provider se ha configurado con una transformación de datos y va a enviar un mensaje a un gestor de colas (por ejemplo si es un Servicio unidireccional o bidireccional que ha recibido un HTTP POST que contiene JSON), MQ Service Provider realiza las siguientes pasos:

1. Toma la carga útil JSON de la solicitud HTTP.
2. Pasa la carga útil a z/OS Connect para realizar la transformación de datos de JSON para una matriz de bytes.
3. Toma el resultado de la transformación de datos y lo envía al gestor de colas en forma de mensaje de bytes de JMS.

De forma predeterminada, el mensaje que se envía tiene el campo MQMD **Format** establecido en MQSTR, lo que indica que el mensaje es un mensaje de serie. Si este no es el caso, puede establecer el atributo de `mqmdFormat` en el elemento `mqzOSConnectService` en un valor apropiado.

El soporte de transformación de datos incorporados siempre genera salida con un CCSID de 37. Esta información debe especificarse en el campo `CodedCharSetId` de MQMD, de lo contrario, es posible que la aplicación que se obtiene de la cola no pueda decodificar el mensaje. Puede hacerlo estableciendo el atributo `CCSID` en la cola referenciada por el elemento de servicio `mqzOSConnectService`.

La configuración de ejemplo siguiente ilustra la configuración adecuada para un servicio unidireccional, que se va a utilizar para enviar mensajes a una cola denominada `SampleQ1`.

Los mensajes deben enviarse con un campo `Format` de MQMD de `AFORMAT` y un campo `CodedCharSetId` de 37.

El elemento `zosConnectDataXform` indica a IBM z/OS Connect EE dónde localizar la configuración para transformaciones de datos y se hace referencia a la utilización del atributo `dataXformRef` del elemento `zosConnectService`.

Para conocer la configuración para un servicio bidireccional, consulte [“Recepción de mensajes de IBM MQ”](#) en la página 580.

```
<jmsConnectionFactory
  id="sampleCF1"
  jndiName="jms/sampleCF1"
  connectionManagerRef="sampleCF1ConnectionManager">

  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21" />
</jmsConnectionFactory>

<connectionManager
  id="sampleCF1ConnectionManager"
  maxPoolSize="5" />

<jmsQueue id="sampleQ1"
  jndiName="jms/sampleQ1">

  <properties.wmqJms
    baseQueueName="SampleQ1"
    CCSID="37" />
</jmsQueue>

<zosConnectService
```

```

id="samplezOSConnectService1"
invokeURI="/samplezOSConnectService1"
serviceName="samplezOSConnectService1_name"
serviceRef="samplezOSConnectService1_MQ"
dataXformRef="xformJSON2Byte"/>

<mqzOSConnectService
  id="samplezOSConnectService1_MQ"
  connectionFactory="jms/sampleCF1"
  mqmdFormat="AFORMAT"
  destination="jms/sampleQ1" />

<zosConnectDataXform id="xformJSON2Byte"
  bindFileLoc="/XFORM_ROOT/bindfiles" bindFileSuffix=".bnd"
  requestSchemaLoc="/XFORM_ROOT/json" requestSchemaSuffix=".json"
  responseSchemaLoc="/XFORM_ROOT/json"
  responseSchemaSuffix=".json" />

```

## Recepción de mensajes de IBM MQ

Cuando una instancia de MQ Service Provider se configura con una transformación de datos y va a recibir un mensaje de un gestor de colas (por ejemplo si se trata de un servicio unidireccional que acciona un HTTP GET o DELETE, o un servicio bidireccional que ha recibido un HTTP POST), MQ Service Provider realiza los pasos siguientes.



**Atención:** Si está utilizando un servicio bidireccional, el servicio ya habrá realizado los pasos descritos en [“Envío de mensajes a IBM MQ” en la página 579.](#)

1. Obtiene el mensaje de la cola.
2. Comprueba que el mensaje sea un JMS **BytesMessage** o un JMS **TextMessage**. Si el mensaje no es ninguno de los dos, se genera un error que se vuelve al interlocutor.
3. Pasa la carga útil de mensaje a z/OS Connect para realizar la transformación de datos de una matriz de bytes a JSON.
4. Toma el resultado de la transformación de datos y lo devuelve como respuesta del método HTTP.

Dependiendo del tipo de mensaje recibido, es posible que necesite realizar una configuración adicional. Esto se debe a que MQ Service Provider debe convertir la carga útil de mensaje recibida en el formato correcto para pasar a los transformación de datos configurada.

La transformación de datos predeterminada espera que la carga útil esté en CCSID 37, sin embargo es posible que el activo de z/OS no esté generando mensajes en este CCSID.

La configuración necesaria depende de si se recibe **BytesMessage** o **TextMessage**, y de si está utilizando z/OS Connect V1, o z/OS Connect EE.

### BytesMessage recibido en IBM z/OS Connect EE

Si se debe recibir un **BytesMessage**, puede especificar los atributos **receiveConversion="QMGR"** y **receiveCCSID="37"** en la definición de cola que se utiliza para recibir el mensaje.

Esto se ilustra en el ejemplo siguiente. En este caso, la definición de sampleQ2Receive tiene los dos atributos **receiveConversion** y **receiveCCSID** establecidos.

```

<jmsConnectionFactory
  id="sampleCF2"
  jndiName="jms/sampleCF2"
  connectionManagerRef="sampleCF2ConnectionManager">

  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21" />
</jmsConnectionFactory>

<connectionManager
  id="sampleCF2ConnectionManager"
  maxPoolSize="5"/>

```

```

<jmsQueue id="sampleQ2Send"
  jndiName="jms/sampleQ2Send">

  <properties.wmqJms
    baseQueueName="SampleQ2Send"
    CCSID="37" />
</jmsQueue>

<jmsQueue id="sampleQ2Receive"
  jndiName="jms/sampleQ2Receive">

  <properties.wmqJms
    baseQueueName="SampleQ2Receive"
    receiveCCSID="37"
    receiveConversion="QMGR" />
</jmsQueue>

<zoscconnect_zosConnectService
  id="samplezOSConnectService2"
  invokeURI="/samplezOSConnectService2"
  serviceName="samplezOSConnectService2_name"
  serviceRef="samplezOSConnectService2_MQ"
  dataXformRef="xformJSON2Byte" />

<mqzoscconnect_mqzOSConnectService
  id="samplezOSConnectService2_MQ"
  connectionFactory="jms/sampleCF2"
  mqmdFormat="AFORMAT"
  destination="jms/sampleQ2Send"
  replyDestination="jms/sampleQ3Receive" />

<zoscconnect_zosConnectDataXform id="xformJSON2Byte"
  bindFileLoc="/XFORM_ROOT/bindfiles" bindFileSuffix=".bnd"
  requestSchemaLoc="/XFORM_ROOT/json" requestSchemaSuffix=".json"
  responseSchemaLoc="/XFORM_ROOT/json"
  responseSchemaSuffix=".json" />

```

## TextMessage recibido en IBM z/OS Connect EE

Si se va a recibir **TextMessage**, tendrá que convertir el mensaje al CCSID previsto por la transformación de datos, de forma predeterminada 37.

Si utiliza una transformación de datos personalizada y la transformación espera un CCSID distinto de 37, puede especificar el atributo `receiveTextCCSID` en el elemento `mqzOSConnectService` para establecer un CCSID adecuado.

Esto se ilustra en el ejemplo siguiente. En este caso, la definición de `zOSConnectService3_MQ` de ejemplo tiene el atributo `receiveTextCCSID` en el atributo de elemento `mqzOSConnectService` establecido en 1208 (UTF-8).

```

<jmsConnectionFactory
  id="sampleCF3"
  jndiName="jms/sampleCF3"
  connectionManagerRef="sampleCF3ConnectionManager">

  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21" />
</jmsConnectionFactory>

<connectionManager
  id="sampleCF3ConnectionManager"
  maxPoolSize="5" />

<jmsQueue id="sampleQ3Send"
  jndiName="jms/sampleQ3Send">

  <properties.wmqJms
    baseQueueName="SampleQ3Send"
    CCSID="37" />
</jmsQueue>

<jmsQueue id="sampleQ3Receive"
  jndiName="jms/sampleQ3Receive">

  <properties.wmqJms

```

```

    baseQueueName="SampleQ3Receive"/>
</jmsQueue>

<zosconnect_zosConnectService
  id="samplezOSConnectService3"
  invokeURI="/samplezOSConnectService3"
  serviceName="samplezOSConnectService3_name"
  serviceRef="samplezOSConnectService3_MQ"
  dataXformRef="customDataXForm"/>

<mqzosconnect_mqzOSConnectService
  id="samplezOSConnectService3_MQ"
  connectionFactory="jms/sampleCF3"
  mqmdFormat="AFORMAT"
  destination="jms/sampleQ3Send"
  replyDestination="jms/sampleQ3Receive"
  receiveTextCCSID="1208" />

```

## Trabajar con aplicaciones MQI

MQ Service Provider proporciona una interfaz RESTful a las aplicaciones existentes que utilizan IBM MQ. Estas aplicaciones pueden interactuar con IBM MQ, utilizando las IBM MQ classes for JMS o la interfaz de cola de mensajes (MQI)

De forma predeterminada, IBM MQ classes for JMS envía mensajes que incluyen una cabecera MQRFH2. Sin embargo, la mayoría de las aplicaciones MQI no utilizan cabeceras MQRFH2.

Si una aplicación IBM MQ classes for JMS interactúa con una aplicación MQI que no se ha diseñado para trabajar con cabeceras MQRFH2, la configuración es necesaria para impedir que las IBM MQ classes for JMS envíen mensajes que contienen una cabecera MQRFH2. Consulte [Correlación de mensajes JMS con mensajes IBM MQ](#) para obtener información adicional.

Si MQ Service Provider va a enviar mensajes a las colas, que serán consumidas por aplicaciones MQI, que no esperan una cabecera MQRFH2, tendrá que configurar IBM z/OS Connect EE para evitar que se envíe una cabecera MQRFH2.

Logre esta configuración añadiendo un atributo **targetClient="MQ"** a la cola de IBM MQ Messaging Provider pertinente en `server.xml`.

La configuración de ejemplo siguiente muestra un servicio de MQ Service Provider unidireccional con un **id** de `mqiService`. A su vez, esto se ha configurado para utilizar la cola de proveedor de mensajería de MQ con un **id** de `mqiQueue`.

`mqiQueue` se ha configurado con el atributo **targetClient="MQ"**. Esto significa que cuando una aplicación envía un mensaje JMS a esa cola (por ejemplo si una aplicación emite un HTTP POST para `mqiService`) no se añadirá una cabecera MQRFH2 al mensaje.

### IBM z/OS Connect EE

```

<mqzosconnect_mqzOSConnectService
  id="mqiService"
  connectionFactory="jms/mqiCF"
  destination="jms/mqiQueue"/>

<jmsQueue
  id="mqiQueue"
  jndiName="jms/mqiQueue">

  <properties.wmqJms
    baseQueueName="MQIQueue"
    targetClient = "MQ"/>
</jmsQueue>

```

## Administración y configuración de IBM MQ Internet Pass-Thru

Para configurar IBM MQ Internet Pass-Thru (MQIPT) realice los cambios en el archivo de configuración `mqipt.conf`.

**Nota:** debe establecer los permisos de archivo seguros en el directorio donde se encuentra `mqipt.conf` para evitar que usuarios no autorizados vean las contraseñas de MQIPT almacenadas o cambien la configuración. Proteja todas las contraseñas especificadas en el archivo de configuración siguiendo el procedimiento indicado en [Cifrado de contraseñas almacenadas](#).

Puede editar el archivo de configuración `mqipt.conf` utilizando IPT Administration Client o un editor de texto de su elección y mandatos de línea de mandatos. Para obtener una lista completa de las propiedades que se pueden especificar en el archivo de configuración `mqipt.conf`, consulte [Referencia de configuración de IBM MQ Internet Pass-Thru](#).

## Iniciar y detener MQIPT

Puede iniciar MQIPT desde la línea de mandatos o hacer que se inicie automáticamente cuando se inicie el sistema. Puede detener MQIPT utilizando IPT Administration Client o la línea de mandatos.

### Inicio de MQIPT desde la línea de mandatos

MQIPT se instala en un directorio de instalación, por ejemplo:

- **Windows** C:\MQIPT en sistemas Windows, con scripts ejecutables en C:\MQIPT\bin
- **Linux** **UNIX** /opt/mqipt en sistemas UNIX and Linux, con scripts ejecutables en /opt/mqipt/bin

MQIPT también utiliza un directorio inicial, que contiene el archivo de configuración `mqipt.conf` y los archivos de salida de MQIPT cuando se está ejecutando. Los subdirectorios siguientes del directorio de inicio de MQIPT se crean automáticamente cuando se invoca MQIPT por primera vez:

- Un directorio `errors` en el que se escriben los archivos First Failure Support Technology (FFST) y de rastreo
- Un directorio `logs` en el que se mantiene el registro de conexión

El ID de usuario con el que se ejecuta MQIPT debe tener permiso para crear estos directorios o bien los directorios ya deben existir y el ID de usuario debe tener permiso para crear, leer y escribir archivos en ellos. Además, si utiliza una política de Java security manager la política de seguridad debe conceder los permisos necesarios para estos directorios. Para obtener más información sobre los valores de política de Security Manager, consulte [Java security manager](#).

Puede utilizar el directorio de instalación como un directorio de inicio. Si utiliza este directorio, debe asegurarse de que el ID de usuario con el que se ejecuta MQIPT tiene los permisos adecuados y que todas las políticas de Security Manager se hayan configurado correctamente.

Para iniciar MQIPT, especifique el mandato siguiente:

- **Windows** En sistemas Windows:

```
MQIPT_INSTALLATION_PATH\bin\mqipt MQIPT_HOME_DIR [-sf encryption_key_file]
```

- **Linux** **UNIX** En sistemas UNIX and Linux:

```
MQIPT_INSTALLATION_PATH/bin/mqipt MQIPT_HOME_DIR [-sf encryption_key_file]
```

Se pueden especificar los siguientes parámetros:

#### DIR\_INICIO\_MQIPT

El directorio de inicio de MQIPT, donde se encuentra el archivo de configuración `mqipt.conf`. Tenga en cuenta que el directorio de inicio de MQIPT se puede especificar como una vía de acceso absoluta o relativa al directorio de trabajo actual del shell de mandatos.

## V 9.1.5 **-sf archivo\_claves\_cifrado**

El nombre de un archivo que contiene la clave de cifrado de contraseña. MQIPT utiliza la clave de cifrado para acceder a las contraseñas cifradas del archivo de configuración `mqipt.conf`.

Este parámetro es opcional. Para obtener información sobre el archivo de claves de cifrado de contraseña y otros métodos para especificar el nombre de archivo al iniciar MQIPT, consulte [“Especificación de la clave de cifrado de contraseña”](#) en la página 584.

Para obtener más información sobre el cifrado de contraseñas almacenadas para ser utilizadas por MQIPT, consulte [Cifrado de contraseñas almacenadas](#).

Los mensajes de la consola muestran el estado de MQIPT. Si se produce un error, consulte [Resolución de problemas de IBM MQ Internet Pass-Thru](#). Los mensajes siguientes son un ejemplo de la salida cuando MQIPT se inicia correctamente:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.1.4.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1414 is starting and will forward messages to :
MQCPI034 ...examplehost(1414)
MQCPI035 ...using MQ protocols
MQCPI057 ...trace level 5 enabled
MQCPI078 Route 1414 ready for connection requests
```

## Inicio automático de MQIPT

Para iniciar MQIPT automáticamente cuando se inicia el sistema, utilice el mandato **mqiptService** para instalar el servicio MQIPT.

- **Windows** En los sistemas Windows, consulte [“Inicio automático de MQIPT en Windows”](#) en la página 585.
- **Linux** **UNIX** En otras plataformas, consulte [“Inicio automático de MQIPT en UNIX o Linux”](#) en la página 586.

## Detener MQIPT

Puede detener MQIPT utilizando IPT Administration Client o la línea de mandatos.

- Para utilizar IPT Administration Client, **mqiptGui**, consulte [“Utilización de IPT Administration Client”](#) en la página 588.
- Para utilizar la herramienta de administración de línea de mandatos, **mqiptAdmin**, consulte [“administrar MQIPT utilizando la línea de mandatos”](#) en la página 587.

Antes de poder detener MQIPT utilizando cualquiera de estos métodos, debe configurar un puerto de mandatos especificando un valor para la propiedad **CommandPort** en el archivo de configuración `mqipt.conf` y habilitar la conclusión remota estableciendo la propiedad **RemoteShutDown** en `true`. Revise las configuraciones de seguridad en [Otras consideraciones de seguridad](#) antes de habilitar el puerto de mandatos de MQIPT o de permitir la conclusión remota.

## V 9.1.5 **Especificación de la clave de cifrado de contraseña**

A partir de IBM MQ 9.1.5, si la configuración de MQIPT contiene contraseñas cifradas mediante una clave de cifrado distinta de la clave predeterminada, debe proporcionar la clave de cifrado de contraseña en un archivo que MQIPT pueda leer al iniciarse.

## Archivo de claves de cifrado de contraseña

Las contraseñas cifradas para almacenarlas y utilizarlas en MQIPT se pueden cifrar mediante una clave de cifrado que proporcione. Si no proporciona ninguna clave de cifrado, se utilizará la clave de cifrado



predeterminada. No tiene que especificar una clave de cifrado de contraseña, pero es más seguro hacerlo. Si no especifica ninguna clave de cifrado, se utilizará la clave de cifrado predeterminada.

Si proporciona una clave de cifrado de contraseña, se debe almacenar en un archivo al que se pueda acceder mediante el mandato **mqiPTPW** que se utiliza para cifrar las contraseñas e MQIPT. Las únicas restricciones en el contenido del archivo son que debe contener al menos un carácter y solo una línea de texto.

**Nota:** debe asegurarse de que se han establecido los permisos de archivo adecuados en el archivo de claves de cifrado de contraseña para impedir que los usuarios no autorizados lean la clave de cifrado. Solo el usuario que ejecuta el mandato **mqiPTPW** y el usuario con el que se ejecuta MQIPT necesitan autorización para leer la clave de cifrado de contraseña.

La misma clave de cifrado de contraseña se utiliza para cifrar y descifrar todas las contraseñas almacenadas para una instancia de MQIPT. Por lo tanto, solo necesita un único archivo de claves de cifrado de contraseña para cada instalación de MQIPT.

Si cambia la clave de cifrado de contraseña de una instalación de MQIPT, se deben volver a cifrar todas las contraseñas cifradas utilizando la nueva clave de cifrado.

## Iniciar MQIPT

El nombre predeterminado del archivo de claves de cifrado de contraseña es *MQIPT\_HOME\_DIR/mqiPT\_cred.key*, donde *MQIPT\_HOME\_DIR* es el directorio donde se almacena el archivo de configuración de *mqiPT.conf*. Si tiene previsto ejecutar MQIPT como un servicio que se inicia automáticamente, debe crear el archivo de claves de cifrado de contraseña con el nombre predeterminado.

Si el archivo de claves de cifrado de contraseña se crea con un nombre que no sea el nombre predeterminado, se debe proporcionar el nombre del archivo a MQIPT al iniciarse. El nombre del archivo de claves de cifrado de contraseña se puede especificar utilizando cualquiera de los métodos siguientes, en orden de preferencia:

1. El parámetro **-sf** en el mandato **mqiPT** utilizado para iniciar MQIPT.
2. la variable de entorno `MQS_MQIPTCRED_KEYFILE`.
3. la propiedad `com.ibm.mq.ipT_cred.keyfile` Java.

Si no se proporciona ningún nombre de archivo de claves de cifrado de contraseña, se utilizará el nombre de archivo predeterminado, si el archivo existe. Si no existe el archivo de claves de cifrado de contraseña predeterminado, se utilizará la clave de cifrado de contraseña predeterminada.

## Windows Inicio automático de MQIPT en Windows

Se proporciona un programa de control de servicio aparte, **mqiPTService.exe**, para que pueda gestionar e iniciar MQIPT como un servicio Windows.

Puede instalar solo un servicio MQIPT en un sistema, aun cuando haya más de una instalación de MQIPT en el sistema.

Debe ejecutar el mandato **mqiPTService** desde un indicador de mandatos de administrador para asegurarse de que dispone de la autorización necesaria para configurar los servicios Windows. Para prepararse para ejecutar el mandato **mqiPTService**, normalmente es necesario realizar estos pasos:

1. Abra un indicador de mandatos de administrador.
2. En el símbolo del sistema, cámbiese de directorio al directorio de instalación de MQIPT, por ejemplo:

```
cd /D C:\mqipt\bin
```

3. Ejecute **mqiPTService.exe** desde este símbolo del sistema.

**mqiPTService.exe** toma los siguientes argumentos de línea de mandatos:

## **mqiptService -install ubicación\_mqipt**

Instala y registra el servicio, de modo que aparece en el panel de servicios de Windows como un servicio automático.

Debe reiniciar Windows después de instalar este servicio.

El parámetro de vía de acceso, que debe proporcionarse, es la vía de acceso completa del directorio que contiene el archivo de configuración `mqipt.conf`. Encierre la vía de acceso entre comillas dobles (") si contiene espacios.

**V 9.1.5** Si la configuración de MQIPT contiene contraseñas que se han cifrado utilizando una clave de cifrado que no sea el valor predeterminado, la clave de cifrado de contraseña debe estar presente en un archivo denominado `mqipt_cred.key` en el mismo directorio que el archivo `mqipt.conf`. Para obtener más información sobre el cifrado de contraseñas en la configuración de MQIPT, consulte [Cifrado de contraseñas almacenadas](#).

## **mqiptService -remove**

Elimina el servicio de modo que ya no aparece en el panel de servicios de Windows.

**Nota:** solo se puede utilizar la instalación de MQIPT que instaló el servicio para eliminarlo. Por ejemplo, si tiene dos instalaciones de MQIPT, una en `C:\mqipt1` y otra en `C:\mqipt2`, y ejecuta el mandato `C:\mqipt1\bin\mqiptService -install C:\mqipt1`, sólo se puede utilizar posteriormente el mandato **mqiptService** de la instalación de `C:\MQIPT1` para eliminar el servicio. Si se intenta eliminar el servicio utilizando otra instalación se produce el error MQCPE083.

## **mqiptService ?**

Muestra los mensajes de ayuda que enumeran los argumentos válidos.

Los parámetros **-install** y **-remove** no se pueden especificar juntos en el mismo mandato.

Si llama al programa `mqiptService.exe` desde la línea de mandatos sin argumentos, el programa supera el tiempo de espera y cede el control con un error.

Cuando se inicia el servicio MQIPT, se inician todas las rutas de MQIPT activas. Cuando se detiene el servicio, todas las rutas están sujetas a la conclusión inmediata.

## **Linux → UNIX Inicio automático de MQIPT en UNIX o Linux**

Se proporciona un programa de control de servicio aparte, **mqiptService**, para que pueda gestionar e iniciar MQIPT como un servicio del sistema UNIX o Linux `init.d` que se inicia cuando el sistema arranca.

Puede instalar solo un servicio MQIPT en un sistema, aun cuando haya más de una instalación de MQIPT en el sistema.

Debe ejecutar el mandato **mqiptService** como usuario `root` para asegurarse de que tiene la autorización necesaria para configurar los servicios.

**mqiptService** toma los siguientes argumentos de línea de mandatos:

## **mqiptService -install**

Instala y registra el servicio.

El archivo `mqipt.conf` para el servicio debe estar ubicado en el directorio de instalación de MQIPT de nivel superior de la instalación desde la que se ha ejecutado **mqiptService**.

**V 9.1.5** Si la configuración de MQIPT contiene contraseñas que se han cifrado utilizando una clave de cifrado que no sea el valor predeterminado, la clave de cifrado de contraseña debe estar presente en un archivo denominado `mqipt_cred.key` en el mismo directorio que el archivo `mqipt.conf`. Para obtener más información sobre el cifrado de contraseñas en la configuración de MQIPT, consulte [Cifrado de contraseñas almacenadas](#).

La instalación del servicio no lo inicia automáticamente. El servicio se inicia la próxima vez que se reinicia el sistema. Consulte la documentación del servicio del sistema operativo si tiene que iniciar el servicio MQIPT de inmediato sin reiniciar.

## mqiptService -remove

Elimina el servicio de modo que ya no se inicia en el tiempo de arranque del sistema.

**Nota:** solo se puede utilizar la instalación de MQIPT que instaló el servicio para eliminarlo. Por ejemplo, si tiene dos instalaciones de MQIPT, una en /opt/mqipt y otra en /usr/local/mqipt, y ejecuta el mandato /opt/mqipt/bin/mqiptService -install, sólo se puede utilizar posteriormente el mandato **mqiptService** de la instalación de /opt/mqipt para eliminar el servicio. Si se intenta eliminar el servicio utilizando otra instalación se produce el error MQCPE083.

## mqiptService ?

Muestra los mensajes de ayuda que enumeran los argumentos válidos.

Los parámetros **-install** y **-remove** no se pueden especificar juntos en el mismo mandato.

Cuando se inicia el servicio MQIPT, se inician todas las rutas de MQIPT activas. Cuando se detiene el servicio, todas las rutas están sujetas a la conclusión inmediata.

## administrar MQIPT utilizando la línea de mandatos

Puede utilizar la línea de mandatos para administrar y configurar MQIPT.

Utilizando un editor de su elección, modifique el archivo de configuración, `mqipt.conf`, para satisfacer sus requisitos. Consulte [Referencia de configuración de IBM MQ Internet Pass-Thru](#) para obtener una lista de las propiedades que puede modificar.

`mqiptAdmin` se puede utilizar para detener una instancia de MQIPT o para renovar una instancia de MQIPT después de realizar cambios en el archivo de configuración. El script `mqiptAdmin` se encuentra en el subdirectorio `bin` del directorio de instalación de MQIPT.

MQIPT está a la escucha de mandatos del script `mqiptAdmin` en el puerto de mandatos si se especifica un valor para la propiedad **CommandPort** en la sección global del archivo de configuración `mqipt.conf`.

**V 9.1.4** Esta propiedad no se especifica en el archivo de configuración de ejemplo proporcionado con MQIPT de IBM MQ 9.1.4.

Para enviar un mandato de renovación a MQIPT, emita el mandato siguiente:

```
mqiptAdmin -refresh {hostname {port} }
```

Para enviar un mandato de detención a MQIPT, emita el mandato siguiente:

```
mqiptAdmin -stop {hostname {port} }
```

*nombre\_host* es el nombre de host del sistema en el que se está ejecutando MQIPT. *puerto* es el puerto de mandatos de MQIPT.

Si no ha proporcionado valores para *nombre\_host* y *puerto*, *nombre\_host* toma como valor predeterminado `localhost` y *puerto* toma como valor predeterminado `1881`.

## refresh

MQIPT vuelve a leer `mqipt.conf` y lleva a cabo las acciones siguientes:

- Si algunas de las rutas activas actualmente están marcadas como inactivas (o ya no se especifican), MQIPT cierra esa ruta y deja de escuchar las conexiones entrantes en ellas.
- Todas las rutas marcadas como activas en el archivo de configuración que no está en ejecución actualmente, las inicia.
- Si los parámetros de configuración de una ruta que se está ejecutando actualmente han cambiado, MQIPT aplica los valores modificados a esas rutas. Siempre que sea posible (por ejemplo, un cambio en el valor de rastreo), lo hace sin interrupciones en la ejecución de las conexiones. Para algunos cambios de parámetro (por ejemplo, un cambio en un destino), MQIPT tiene que cerrar todas las conexiones antes de efectuar el cambio y reiniciar la ruta.

## stop

MQIPT cierra todas las conexiones, deja de escuchar las conexiones entrantes y luego sale.

Este mandato se ignora a menos que el archivo `mqipt.conf` especifique `RemoteShutDown=true`.

**Nota:** En los sistemas Windows, estas funciones administrativas también están disponibles en el menú **Inicio > Programas** si se crearon los iconos de menú Inicio de MQIPT cuando se instaló MQIPT.

## Utilización de IPT Administration Client

IPT Administration Client es una interfaz gráfica de usuario que muestra las propiedades globales y las propiedades específicas de ruta para cada instancia de MQIPT. Puede utilizar IPT Administration Client para configurar y actualizar una o varias instancias de MQIPT.

**Nota:** IPT Administration Client está en desuso. Es posible que las versiones actuales de IPT Administration Client no funcionen con versiones futuras de MQIPT. Para obtener un método alternativo de configuración y administración de MQIPT, consulte [Administración de MQIPT utilizando la línea de mandatos](#).

Los únicos datos almacenados localmente por IPT Administration Client son la lista de instancias de MQIPT, en un archivo llamado `client.conf`. Las propiedades globales y de ruta siempre se recuperan de MQIPT antes de que se muestren en IPT Administration Client. Por lo tanto, siempre se ve el estado actual de cada instancia de MQIPT.

## Inicio de IPT Administration Client

Para iniciar IPT Administration Client utilice el script `mqiptGui` que se encuentra en el subdirectorio MQIPT `bin`.

La primera vez que se inicia IPT Administration Client, se le solicita información de conexión para una instancia de MQIPT. Debe indicar la información siguiente:

### Nombre de MQIPT

Un nombre de su elección que se utilizará para describir esta instancia de MQIPT.

### Dirección de red

La dirección del sistema en el que está instalada esta instancia de MQIPT. La dirección puede ser un nombre reconocido por el servidor de nombres, una dirección decimal con puntos o el sistema principal local (si esta instancia se encuentra en el mismo sistema que IPT Administration Client).

### Puerto de mandatos

El número del puerto en el que está a la escucha esta instancia de MQIPT.

### Tiempo de espera (seg.)

El número de segundos que IPT Administration Client espera a una conexión a esta instancia de MQIPT. Mantenga este valor tan bajo como sea posible para reducir el tiempo de renovación.

### Contraseña de acceso

La contraseña utilizada al comunicarse con esta instancia de MQIPT. Rellene este campo solo si está vigente la comprobación de contraseñas. (La comprobación de contraseñas está vigente si **AccessPW** está establecida en la sección de propiedades globales del archivo de configuración `mqipt.conf` y tiene un valor que no sea una serie vacía).

### Guardar contraseña

Seleccione el recuadro de selección **Guardar contraseña** para guardar la contraseña localmente para las sesiones futuras. Si se borra **Guardar contraseña**, la contraseña se guarda solo durante la sesión actual o hasta que se elimine esta instancia de MQIPT en IPT Administration Client).

La instancia de MQIPT debe estar en ejecución y a la escucha en el puerto de mandatos para que IPT Administration Client pueda conectarse a ella.

**V 9.1.4** La propiedad **CommandPort** se debe especificar en la sección global del archivo de configuración `mqipt.conf` para habilitar el puerto de mandatos. Esta propiedad no se especifica en el archivo de configuración de ejemplo que se proporciona con MQIPT desde IBM MQ 9.1.4.

## Administración de una instancia de MQIPT

Puede ver y actualizar las propiedades globales y de ruta de las instancias de MQIPT utilizando IPT Administration Client.

Consulte [“Inicio de IPT Administration Client”](#) en la página 588 para aprender a iniciar IPT Administration Client.

Seleccione una instancia de MQIPT en la lista para recuperar las propiedades globales y de ruta de esta instancia de MQIPT. Si esta instancia de MQIPT no se está ejecutando o no se ha especificado el valor correcto de **CommandPort** en la sección de propiedades globales del archivo de configuración `mqipt.conf`, se emite un mensaje de error. Para cambiar el nombre de host y el puerto de mandatos, pulse **MQIPT > Conexión**.

Efectúe una doble pulsación en una instancia de MQIPT en la lista para mostrar una lista de rutas disponibles. Seleccione una ruta para cambiar sus propiedades.

Puede añadir una ruta pulsando **MQIPT > Añadir ruta**. Se mostrarán los valores de propiedad predeterminados para la nueva ruta, tal como se define en el conjunto de propiedades globales para esta instancia de MQIPT.

Pulse **MQIPT > Aplicar** para aplicar los cambios realizados. Los cambios se guardan en el archivo de configuración `mqipt.conf` de esta instancia de MQIPT y se producen de forma inmediata.

**Nota:** Las líneas de comentario añadidas en el archivo `mqipt.conf` se pierden cuando se actualiza.

## Establecimiento de las propiedades de MQIPT

El valor que se utiliza para cada propiedad de MQIPT está determinado por el lugar en el que se establece.

1. Si establece el valor de un parámetro en la sección `[route]` del archivo de configuración `mqipt.conf`, ese valor se utiliza para la ruta y altera temporalmente cualquier valor establecido en la sección `[global]`. Los valores de ruta son específicos de una ruta única; no afectan a ninguna otra ruta.
2. Si establece el valor de un parámetro en la sección `[global]` del archivo de configuración `mqipt.conf`, ese valor se utiliza para todas las rutas, a menos que se altere temporalmente de forma explícita para una ruta determinada.
3. Todas las propiedades de MQIPT tienen valores predeterminados que se utilizan si no se establecen de forma explícita en la sección `[global]` o `[route]` del archivo de configuración `mqipt.conf`. Consulte [Resumen de propiedades de MQIPT](#) para obtener una lista de los valores predeterminados.

## Ejemplo

En el ejemplo siguiente se muestra cómo se ven afectados los valores utilizados para los dos parámetros **LDAPCacheTimeout** y **MinConnectionThreads** según el lugar en el que se establecen los parámetros.

Valores del archivo de configuración `mqipt.conf`:

```
[global]
LDAPCacheTimeout = 120

[route]
name = route1
MinConnectionThreads = 10

[route]
name = route2
LDAPCacheTimeout = 60
```

Valores utilizados por MQIPT:

```
route1
LDAPCacheTimeout:      120 (not set in route1 [route] section; set in [global] section)
```

```
MinConnectionThreads: 10 (set in [route] section)
route2
LDAPCacheTimeout: 60 (set in [route] section; [global] section ignored)
MinConnectionThreads: 5 (not set in mqipt.conf for route2; uses the default value)
```

## Opciones de menú de IPT Administration Client

### Menú Archivo

Puede gestionar la lista de instancias de MQIPT utilizando las opciones siguientes que están disponibles en el menú **Archivo**:

#### Añadir MQIPT

Añade una nueva instancia de MQIPT a la lista en IPT Administration Client. Consulte [“Inicio de IPT Administration Client”](#) en la [página 588](#) para obtener información detallada acerca de la información que debe especificar.

#### Eliminar MQIPT

Elimina la instancia resaltada actualmente de MQIPT de la lista en IPT Administration Client. Esta opción no detiene ni afecta a la ejecución de esta instancia de MQIPT.

#### Guardar configuración

Guarda la lista de instancias de MQIPT en el archivo de configuración de IPT Administration Client local para que se puedan restaurar la próxima vez que se inicie IPT Administration Client. Únicamente este MQIPT se guarda localmente; las propiedades `[global]` y `[route]` siempre se recuperan de cada instancia de MQIPT.

#### Abandonar

Detiene IPT Administration Client. Se le da la opción de guardar los cambios pendientes antes de que se cierre IPT Administration Client.

### Menú MQIPT


Puede gestionar la instancia seleccionada de MQIPT utilizando las opciones siguientes que están disponibles en el menú **MQIPT**:

#### Conexión

Cambia las propiedades de acceso de una instancia de MQIPT. Consulte [“Inicio de IPT Administration Client”](#) en la [página 588](#) para obtener información detallada acerca de la información que puede actualizar.

#### Contraseña

Cambia la contraseña necesaria para acceder a una instancia de MQIPT. Deje en blanco el campo **Contraseña actual** si no hay ninguna contraseña establecida actualmente. No especifique una nueva contraseña si desea dejar de utilizar una contraseña. Seleccione el recuadro de selección **Guardar contraseña** si desea guardar la contraseña localmente. Si no guarda la contraseña, debe especificarla cada vez que desee acceder a esta instancia de MQIPT.

 No puede cambiar la contraseña para una instancia de MQIPT que se ha configurado con una contraseña de acceso cifrada.

#### Añadir ruta

Añade una ruta a una instancia seleccionada de MQIPT. Cada ruta debe tener un puerto de escucha exclusivo para una instancia de MQIPT.

#### Suprimir ruta

Suprime la ruta seleccionada de la instancia de MQIPT. La supresión no surte efecto hasta que no se aplica, pulsando **MQIPT > Aplicar**.

#### Aplicar

Actualiza el archivo de configuración de una instancia de MQIPT. Los nuevos valores entran en vigor de forma inmediata.

## Renovar

Lee el archivo de configuración actual de la instancia seleccionada de MQIPT y renueva la pantalla.

## Detener

Detiene una instancia de MQIPT que se está ejecutando. Después de este mandato, pierde el contacto con MQIPT. Este mandato se ignora a menos que se active la propiedad global RemoteShutdown.

Las propiedades de ruta se pueden actualizar de la misma manera que las propiedades globales de MQIPT. Cuando cambie cualquier propiedad de una ruta, debe aplicar los cambios para que entren en vigor. Puede hacerlo seleccionando la opción de menú **MQIPT > Aplicar** o respondiendo Yes cuando se le solicite que guarde la configuración.

## Información sobre la accesibilidad

La interfaz gráfica de usuario de IPT Administration Client se ha creado teniendo en cuenta la accesibilidad. Es sencillo realizar todas las funciones disponibles sin utilizar ratón, mediante el uso de equivalentes de teclado. Puede navegar por la pantalla utilizando las teclas Tab, Mayús + Tab, Ctrl + Tab y las teclas de flecha arriba, flecha abajo, flecha izquierda y flecha derecha de la forma estándar. Puede obtener el equivalente para pulsar los botones seleccionando primero el botón y luego pulsando Intro.

Puede acceder a las opciones de menú combinando las teclas de tabulador y de flecha o bien utilizando las teclas aceleradoras, que están disponibles para todas las opciones. Por ejemplo, puede cerrar la GUI seleccionando primero Alt + f y, a continuación, Alt + q (Archivo- > Salir). Cuando llega a un elemento de menú, puede activarlo utilizando la tecla Intro.

Puede utilizar las teclas de flecha para navegar por el árbol. En particular, puede utilizar las teclas de flecha derecha e izquierda para abrir o cerrar un nodo de MQIPT, lo que le permite mostrar u ocultar las rutas.

Puede utilizar la tecla de barra espaciadora para cambiar el estado de las casillas de verificación seleccionadas. Puede utilizar la tecla Intro para seleccionar los campos para editarlos.

## Aspecto

Idealmente, la interfaz gráfica de usuario de IPT Administration Client debe adoptar el aspecto del entorno. Puesto que esto no siempre es posible, puede proporcionar un archivo de configuración para adaptar el aspecto de la interfaz gráfica de usuario para que se ajuste a sus necesidades. El archivo de configuración se denomina `custom.properties` y se debe colocar en el subdirectorio `bin`.

Utilice este archivo de configuración para configurar los elementos siguientes de la interfaz gráfica de usuario:

- El color de primer plano (el color del texto)
- El color de fondo
- La fuente del texto
- El estilo del texto (sin formato, negrita, cursiva o negrita y cursiva)

Se proporciona un archivo de configuración de ejemplo `customSample.properties` en el subdirectorio `samples` de la vía de acceso de instalación de MQIPT, que contiene comentarios que muestran cómo se puede modificar. Se recomienda que copie este archivo en `bin/custom.properties` y realice los cambios necesarios.

## Realización de copias de seguridad

Debe hacer copia de seguridad de una serie de archivos de MQIPT como parte de los procedimientos de copia de seguridad habituales.

Realice copia de seguridad de los archivos siguientes de forma regular:

- El archivo de configuración, `mqipt.conf`
- Los archivos de conjunto de claves SSL/TLS especificados por las siguientes propiedades en `mqipt.conf`:

- **SSLClientKeyRing**
- **SSLClientCAKeyRing**
- **SSLServerKeyRing**
- **SSLServerCAKeyRing**
- Los archivos de contraseña del conjunto de claves SSL/TLS especificados por las siguientes propiedades en `mqipt.conf`:
  - **SSLClientKeyRingPW**
  - **SSLClientCAKeyRingPW**
  - **SSLServerKeyRingPW**
  - **SSLServerCAKeyRingPW**
- **V 9.1.5** El archivo de claves de cifrado de contraseña, si la configuración de MQIPT contiene contraseñas que están cifradas con una clave de cifrado distinta de la clave predeterminada.
- El archivo de configuración de IPT Administration Client, `client.conf`, que contiene información de conexión sobre todas las instancias de MQIPT conocidas para IPT Administration Client.
- El archivo de política especificado en **SecurityManagerPolicy**, si se ha establecido esa propiedad.
- Los archivos de salida de seguridad y los archivos de salida de certificado especificados por las siguientes propiedades en `mqipt.conf`:
  - **SecurityExitName**
  - **SSLExitName**

## Ajuste de rendimiento

Puede ajustar el rendimiento relativo de cada ruta de MQIPT utilizando una combinación de una agrupación de hebras y una especificación de tiempo de espera inactivo.

### Hebras de conexión

A cada ruta de MQIPT se le asigna una agrupación de trabajo de hebras que se ejecutan simultáneamente y gestionan las solicitudes de comunicación entrantes. Durante la inicialización, se crea una agrupación de hebras (del tamaño especificado en el atributo `MinConnectionThreads` de la ruta) y se asigna una hebra para gestionar la primera solicitud entrante. Cuando llega esta solicitud, se asigna otra hebra, lista para la siguiente solicitud entrante. Cuando se han asignado todas las hebras para el trabajo, se crea una nueva hebra, se añade a la agrupación de trabajo y se asigna para el trabajo.

De este modo, la agrupación aumenta hasta que se alcanza el número máximo de hebras (especificado en **MaxConnectionThreads**). Las hebras se liberan de nuevo a la agrupación cuando finaliza una conversación o transcurre el período de tiempo de espera inactivo especificado. Cuando se alcanza el número máximo de hebras de trabajo, la solicitud entrante siguiente espera hasta que se libere una hebra de nuevo a la agrupación de trabajo.

Puede reducir el tiempo que las solicitudes podrían tener que esperar aumentando el número de hebras disponibles. Sin embargo, debe equilibrar este aumento con los recursos del sistema que están disponibles.

### Tiempo de espera inactivo

De forma predeterminada, las hebras de trabajo no terminan debido a la inactividad. Cuando se ha asignado una hebra a una conversación, permanece asignada a esa conversación hasta que se cierra normalmente, se desactiva la ruta o se cierra MQIPT. Opcionalmente, puede especificar un intervalo de intervalo de tiempo de espera inactivo (en minutos) en la propiedad **IdleTimeout** para que las hebras



que han estado inactivas durante el periodo de tiempo especificado se reciclen. Las hebras se reciclan para su uso, colocándolas de nuevo en la agrupación de trabajo.

Si la actividad de IBM MQ es intermitente, establezca su intervalo de pulsaciones en un valor menor que el del tiempo de espera de MQIPT para que las hebras no se reciclen constantemente.



Esta información se ha desarrollado para productos y servicios ofrecidos en los Estados Unidos.

Es posible que IBM no ofrezca los productos, servicios o las características que se tratan en este documento en otros países. Consulte al representante local de IBM para obtener información sobre los productos y servicios disponibles actualmente en su zona. Las referencias a programas, productos o servicios de IBM no pretenden indicar ni implicar que sólo puedan utilizarse los productos, programas o servicios de IBM. En su lugar podrá utilizarse cualquier producto, programa o servicio equivalente que no infrinja ninguno de los derechos de propiedad intelectual de IBM. No obstante, es responsabilidad del usuario evaluar y verificar el funcionamiento de cualquier producto, programa o servicio no IBM.

IBM puede tener patentes o solicitudes de patentes pendientes que cubran el tema principal descrito en este documento. El suministro de este documento no le otorga ninguna licencia sobre estas patentes. Puede enviar consultas sobre licencias, por escrito, a:

IBM Director  
of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

Para consultas sobre licencias relacionadas con información de doble byte (DBCS), póngase en contacto con el Departamento de propiedad intelectual de IBM de su país o envíe las consultas por escrito a:

Licencias de Propiedad Intelectual  
Ley de Propiedad intelectual y legal  
IBM Japan, Ltd.  
19-21, Nihonbashi-Hakozakicho, Chuo-ku  
Tokio 103-8510, Japón

**El párrafo siguiente no se aplica al Reino Unido ni a ningún otro país donde estas disposiciones contradigan la legislación vigente:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROPORCIONA ESTA PUBLICACIÓN "TAL CUAL" SIN NINGÚN TIPO DE GARANTÍA, YA SEA EXPLÍCITA O IMPLÍCITA, INCLUYENDO, PERO SIN LIMITARSE A, LAS GARANTÍAS IMPLÍCITAS DE NO INCUMPLIMIENTO, COMERCIALIZABILIDAD O IDONEIDAD PARA UNA FINALIDAD DETERMINADA. Algunas legislaciones no contemplan la exclusión de garantías, ni implícitas ni explícitas, en determinadas transacciones, por lo que puede haber usuarios a los que no les afecte dicha norma.

Esta información puede contener imprecisiones técnicas o errores tipográficos. La información aquí contenida está sometida a cambios periódicos; tales cambios se irán incorporando en nuevas ediciones de la publicación. IBM puede efectuar mejoras y/o cambios en los productos y/o programas descritos en esta publicación en cualquier momento y sin previo aviso.

Cualquier referencia en esta información a sitios web que no son de IBM se realiza por razones prácticas y de ninguna manera sirve como un respaldo de dichos sitios web. Los materiales de dichos sitios web no forman parte de este producto de IBM y la utilización de los mismos será por cuenta y riesgo del usuario.

IBM puede utilizar o distribuir cualquier información que el usuario le proporcione del modo que considere apropiado sin incurrir por ello en ninguna obligación con respecto al usuario.

Los titulares de licencias de este programa que deseen información del mismo con el fin de permitir: (i) el intercambio de información entre los programas creados de forma independiente y otros programas (incluido este) y (ii) el uso mutuo de la información intercambiada, deben ponerse en contacto con:

IBM Corporation  
Software Interoperability Coordinator, Department 49XA  
3605 Highway 52 N

Rochester, MN 55901  
U.S.A.

Dicha información puede estar disponible, sujeta a los términos y condiciones apropiados, incluyendo, en algunos casos, el pago de una cantidad.

El programa bajo licencia que se describe en esta información y todo el material bajo licencia disponible para el mismo lo proporciona IBM bajo los términos del Acuerdo de cliente de IBM, el Acuerdo de licencia de programas internacional de IBM o cualquier acuerdo equivalente entre las partes.

Los datos de rendimiento incluidos en este documento se han obtenido en un entorno controlado. Por consiguiente, los resultados obtenidos en otros entornos operativos pueden variar de manera significativa. Es posible que algunas mediciones se hayan realizado en sistemas en nivel de desarrollo y no existe ninguna garantía de que estas mediciones serán las mismas en sistemas disponibles generalmente. Además, algunas mediciones pueden haberse estimado por extrapolación. Los resultados reales pueden variar. Los usuarios de este documento deben verificar los datos aplicables a su entorno específico.

La información relativa a productos que no son de IBM se obtuvo de los proveedores de esos productos, sus anuncios publicados u otras fuentes de disponibilidad pública. IBM no ha comprobado estos productos y no puede confirmar la precisión de su rendimiento, compatibilidad o alguna reclamación relacionada con productos que no sean de IBM. Las preguntas relacionadas con las posibilidades de los productos que no sean de IBM deben dirigirse a los proveedores de dichos productos.

Todas las declaraciones relacionadas con una futura intención o tendencia de IBM están sujetas a cambios o se pueden retirar sin previo aviso y sólo representan metas y objetivos.

Este documento contiene ejemplos de datos e informes que se utilizan diariamente en la actividad de la empresa. Para ilustrar los ejemplos de la forma más completa posible, éstos incluyen nombres de personas, empresas, marcas y productos. Todos estos nombres son ficticios y cualquier similitud con los nombres y direcciones utilizados por una empresa real es puramente casual.

#### LICENCIA DE COPYRIGHT:

Esta información contiene programas de aplicación de ejemplo en lenguaje fuente que ilustran técnicas de programación en diversas plataformas operativas. Puede copiar, modificar y distribuir estos programas de ejemplo de cualquier forma sin pagar ninguna cuota a IBM para fines de desarrollo, uso, marketing o distribución de programas de aplicación que se ajusten a la interfaz de programación de aplicaciones para la plataforma operativa para la que se han escrito los programas de ejemplo. Los ejemplos no se han probado minuciosamente bajo todas las condiciones. IBM, por tanto, no puede garantizar la fiabilidad, servicio o funciones de estos programas.

Puede que si visualiza esta información en copia software, las fotografías e ilustraciones a color no aparezcan.

## Información acerca de las interfaces de programación

---

La información de interfaz de programación, si se proporciona, está pensada para ayudarle a crear software de aplicación para su uso con este programa.

Este manual contiene información sobre las interfaces de programación previstas que permiten al cliente escribir programas para obtener los servicios de WebSphere MQ.

Sin embargo, esta información puede contener también información de diagnóstico, modificación y ajustes. La información de diagnóstico, modificación y ajustes se proporciona para ayudarle a depurar el software de aplicación.

**Importante:** No utilice esta información de diagnóstico, modificación y ajuste como interfaz de programación porque está sujeta a cambios.

## Marcas registradas

---

IBM, el logotipo de IBM , ibm.com, son marcas registradas de IBM Corporation, registradas en muchas jurisdicciones de todo el mundo. Hay disponible una lista actual de marcas registradas de IBM en la web en "Copyright and trademark information"[www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml). Otros nombres de productos y servicios pueden ser marcas registradas de IBM o de otras empresas.

Microsoft y Windows son marcas registradas de Microsoft Corporation en EE.UU. y/o en otros países.

UNIX es una marca registrada de Open Group en Estados Unidos y en otros países.

Linux es una marca registrada de Linus Torvalds en Estados Unidos y en otros países.

Este producto incluye software desarrollado por Eclipse Project (<http://www.eclipse.org/>).

Java y todas las marcas registradas y logotipos son marcas registradas de Oracle o sus afiliados.







Número Pieza:

(1P) P/N: